



ECOLE NORMALE SUPÉRIEURE DE L'ENSEIGNEMENT  
TECHNIQUE DE MOHAMMEDIA  
UNIVERSITÉ HASSAN II DE CASABLANCA

**Département Mathématiques et Informatique**  
**Filière : Ingénierie Informatique – Cybersécurité et Confiance Numérique**

## Rapport de Projet d'Innovation

# Infrastructure réseau ultra sécurisé

### Réalisé par :

Hettaba chayma (II-CCN2)  
Mariam Sati (II-CCN2)  
Hachani Wijdane (II-CCN2)  
Houda Allali (II-CCN2)  
Agnaou ilyass (II-CCN2)

### Encadré par :

Pr. Youssfi Mohamed  
Pr. Tadlaoui Ahmed

11 juin 2025

Année universitaire : 2024–2025

ENSET, Avenue Hassan II - B.P. 159 - Mohammedia - Maroc

☎ 05 23 32 22 20 / 05 23 32 35 30 – Fax : 05 23 32 25 46 - Site Web: [www.enset-media.ac.ma](http://www.enset-media.ac.ma)  
E-Mail : [enset-media@enset-media.ac.ma](mailto:enset-media@enset-media.ac.ma)

## Remerciements :

---

Nous tenons à exprimer notre profonde gratitude à nos encadrants, M. Youssfi Mohamed et M. Tadlaoui Ahmed, pour leur accompagnement, leurs conseils avisés et leur disponibilité tout au long de la réalisation de ce projet. Leur expertise et leur soutien ont grandement contribué à la réussite de notre travail.

Nous remercions également chaleureusement notre professeur, M. Azeddine Khiat, qui a su faciliter notre travail en mettant à notre disposition des ressources pédagogiques précieuses et en nous guidant avec patience et bienveillance.

Nos remerciements s'adressent aussi à Pr. Abdelaziz Daaif, dont les orientations et les encouragements ont été d'une grande aide pour mener à bien ce projet.

Enfin, nous remercions toutes les personnes qui, de près ou de loin, ont contribué à la réalisation de ce projet, que ce soit par leur soutien moral, leurs conseils ou leur aide technique.

# Table des matières

---

Introduction :	1
Première partie : Présentation du projet	2
1. Origine du projet :	2
2. Objectifs :	2
3. Cible du projet :	2
4. Architecture global :	4
Deuxième partie : Outils et méthodologie :	7
1. Outils utilisés :	7
2. Méthodologie adoptée	8
a. Conception logique du réseau	8
b. Simulation sous GNS3	8
c. Génération et analyse d'attaques	9
d. Rapport et documentation	9
3. Faisabilité du projet :	9
4. Organisation et mode de travail	13
Troisième partie : Réalisation technique du projet	16
1. Architecture réseau conçue :	16
2. Configuration des équipements réseau :	17
3. Configuration des Serveurs :	29
Test de sécurité	48
1. Tests du WAF (ModSecurity)	48
2. Test de la méthode de détection d'intrusion avec notre SIEM OSSIM.	51
3. Analyse et discussion	52
Conclusion	56
Références bibliographiques :	57

## Introduction :

---

Dans un monde numérique toujours plus interconnecté, la cybersécurité constitue un enjeu majeur pour la protection des données et des infrastructures informatiques. Face à la montée en puissance des cyber menaces — attaques DDoS, intrusions réseau, logiciels malveillants, ou encore fuites de données sensibles — il devient impératif de mettre en place des solutions performantes capables de détecter, analyser et neutraliser efficacement toute activité malveillante.

L'un des défis les plus critiques en matière de sécurité des réseaux réside dans la capacité à détecter de manière proactive les comportements suspects avant qu'ils ne causent des dommages. C'est dans ce contexte que s'inscrit notre projet d'innovation intitulé : « **Infrastructure réseau ultra sécurisé** ».

Ce projet a été mené dans le cadre de notre formation en deuxième année à l'École Nationale Supérieure d'Enseignement Technique (ENSET), au sein de la filière **Ingénierie Informatique – Cybersécurité et Confiance Numérique**. Il a pour objectif de concevoir, simuler et évaluer une architecture réseau sécurisée, intégrant des mécanismes avancés de détection et de prévention des intrusions, en s'appuyant sur différentes approches de surveillance appliquées aux pare-feu.

Notre démarche repose sur une modélisation réaliste du trafic réseau, la simulation de divers scénarios d'attaques, et l'intégration de technologies clés telles que GNS3, ModSecurity, FortiGate et Kali Linux. Ces outils nous ont permis de créer un environnement virtuel représentatif d'une infrastructure professionnelle, où chaque couche de sécurité vient renforcer la résilience globale du système.

À travers cette expérience, nous avons mis en pratique plusieurs concepts fondamentaux de la cybersécurité : segmentation du réseau, détection des intrusions, réponse aux incidents, et mise en œuvre de politiques de filtrage dynamiques. Ce projet constitue ainsi une étape importante dans notre parcours académique, en consolidant nos compétences techniques et en développant une vision opérationnelle de la sécurité des systèmes d'information.

# Première partie : Présentation du projet

---

## 1. Origine du projet :

La multiplication des cyberattaques (DDoS, ransomware, exfiltration de données) place la détection proactive au cœur des enjeux de sécurité. Soucieux d'acquérir une expertise opérationnelle, notre groupe d'étudiants de 2<sup>e</sup> année II-CCN à l'ENSET Mohammedia a lancé le projet « Infrastructure réseau ultra sécurisée » consiste à simuler un réseau informatique sécurisé comportant plusieurs niveaux de défense. Ces systèmes permettent respectivement de détecter et de bloquer les activités suspectes ou malveillantes au sein du réseau à travers :

- Tester plusieurs méthodes de détection d'intrusion et de filtrage au niveau pare-feu.
- Simuler un environnement professionnel complet (LAN, DMZ, Internet) pour évaluer la résilience d'une architecture multicouche.
- Développer une démarche de cybersécurité alignée sur les bonnes pratiques industrielles.

## 2. Objectifs :

- **Objectif principal :**

Concevoir, configurer et évaluer une architecture réseau capable d'identifier et de stopper les attaques courantes avant qu'elles n'affectent les ressources critiques.

- **Objectifs spécifiques :**

1. Mettre en place un pare-feu FortiGate et comparer différentes stratégies de détection (signature, comportementale, règles OWASP CRS, etc.).
2. Sécuriser un serveur Web Apache2 avec ModSecurity dans une DMZ.
3. Élaborer des scénarios d'attaques (port-scanning, injection SQL, XSS) et mesurer la réponse du système.

## 3. Cible du projet :

Ce projet s'adresse principalement aux **petites et moyennes entreprises (PME)**, qui sont de plus en plus exposées aux cybermenaces, mais qui ne disposent pas toujours des ressources techniques, humaines ou financières pour se doter de solutions de sécurité complexes ou onéreuses.

### ➤ **Cible primaire :**

Les PME cherchant à :

- Mettre en place une architecture réseau sécurisée sans recourir à des solutions commerciales coûteuses.
- Protéger leurs services critiques (site web, serveur DNS, FTP, etc.) contre des attaques externes telles que les scans de ports, les intrusions, les injections SQL ou les attaques DDoS.
- Disposer d'une solution reproductible, simple à maintenir et adaptée à leurs moyens.
- Se conformer aux bonnes pratiques en matière de segmentation réseau, de filtrage dynamique, et d'analyse des activités suspectes.

### ➤ **Bénéfices pour cette cible :**

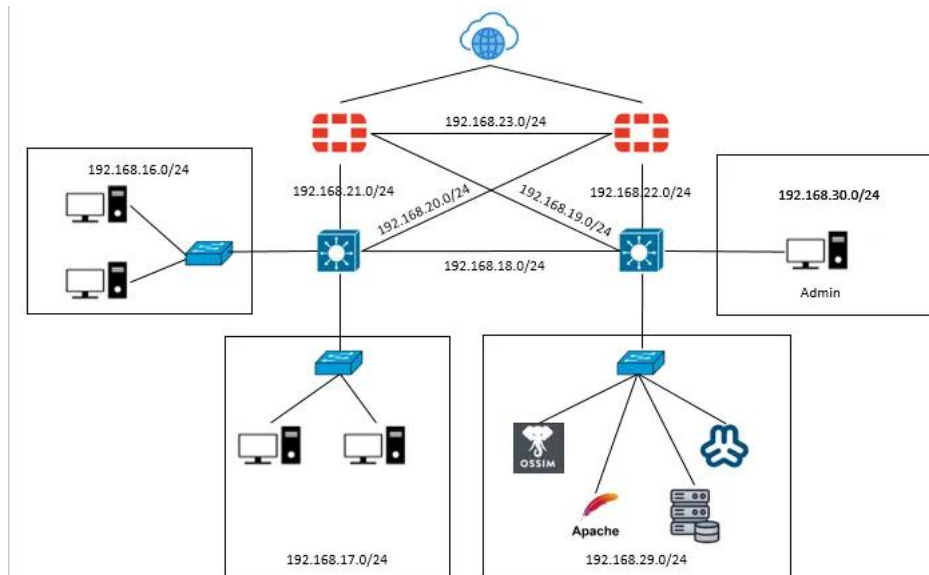
- Accès à une infrastructure de sécurité modulable, facilement à déployer avec des outils open-source ou semi-professionnels (GNS3, ModSecurity, FortiGate en version de démonstration, etc.).
- Acquisition d'un modèle reproductible, adaptable selon la taille du réseau ou les services déployés.
- Réduction des risques de cyberattaques en appliquant des techniques de défense multicouches, allant de la détection d'intrusions à la surveillance du trafic réseau.

### ➤ **Cibles secondaires :**

- Étudiants ou formateurs en cybersécurité, à la recherche d'un projet pédagogique appliqué pour illustrer les concepts de pare-feu, DMZ, segmentation, WAF, etc.
- Startups techniques qui ont besoin de sécuriser leur réseau dès la phase de prototypage, sans investir dans des infrastructures complexes.

## 4. Architecture global :

L'architecture de ce projet consiste à créer un réseau informatique très sécurisé pour l'entreprise. L'objectif principal est de protéger les données et les systèmes tout en permettant aux utilisateurs de travailler efficacement. Cette infrastructure est conçue pour être fiable, sécurisée et facile à gérer.



### Organisation du réseau

#### ➤ Principe de base : la séparation

Le réseau est organisé en zones séparées, chaque zone a un rôle précis et des règles de sécurité adaptées. Cette séparation permet de mieux contrôler qui peut accéder à quoi, et empêche les problèmes de se propager d'une zone à l'autre.

#### ➤ Structures en plusieurs niveaux

Le réseau fonctionne comme un bâtiment à plusieurs étages :

- **Niveau d'entrée** : contrôle tout ce qui vient de l'extérieur
- **Niveau central** : distribue les informations dans tout le réseau
- **Niveau zones** : chaque service a son propre espace

### Les différentes zones du réseau :

#### ➤ Zone d'administration

Cette zone est réservée aux administrateurs informatiques. C'est ici qu'ils gèrent et surveillent tout le réseau. Seules les personnes autorisées peuvent y accéder. On y trouve les outils de contrôle et de surveillance du système.

### ➤ **Zone des utilisateurs**

C'est l'espace de travail normal des employés. Ils y trouvent tout ce dont ils ont besoin pour leur travail quotidien : accès aux applications, partage de fichiers, navigation internet contrôlée. Cette zone est sécurisée mais reste pratique à utiliser.

### ➤ **Zone de développement**

Cette zone est séparée pour les équipes qui créent ou modifient des applications. Elle permet de tester de nouveaux programmes sans risquer d'affecter le travail des autres utilisateurs. C'est un espace d'expérimentation sécurisé.

### ➤ **Zone des serveurs**

C'est le cœur technique du système. On y trouve tous les serveurs importants : ceux qui stockent les données, ceux qui font fonctionner les sites web, et ceux qui gèrent les programmes. Cette zone est la plus protégée car elle contient les éléments les plus critiques.

#### **Points forts de cette architecture :**

##### ➤ **Sécurité renforcée**

- Chaque zone est protégée individuellement
- Si un problème arrive dans une zone, il ne peut pas facilement se propager aux autres
- Plusieurs niveaux de protection sont mis en place
- Tout ce qui se passe sur le réseau est surveillé et enregistré

##### ➤ **Facilité de gestion**

- Les administrateurs peuvent gérer chaque zone séparément
- Il est facile d'ajouter de nouveaux utilisateurs ou services
- Les pannes sont plus faciles à localiser et réparer
- La maintenance peut se faire zone par zone sans arrêter tout le système

##### ➤ **Évolution possible**

- On peut facilement ajouter de nouvelles zones si besoin
- Chaque zone peut évoluer indépendamment des autres
- Le système peut grandir avec l'entreprise
- Les nouvelles technologies peuvent être intégrées progressivement



### **Fonctionnement quotidien :**

#### **➤ Pour les utilisateurs**

Les employés se connectent normalement à leur ordinateur et accèdent aux ressources autorisées. Ils ne voient pas la complexité du système, mais bénéficient de sa sécurité et de sa fiabilité.

#### **➤ Pour les administrateurs**

Ils disposent d'outils centralisés pour surveiller tout le réseau, gérer les accès, et intervenir rapidement en cas de problème. Ils peuvent voir ce qui se passe dans chaque zone et agir en conséquence.

#### **➤ En cas de problème**

Si un incident arrive, il reste localisé dans sa zone. Les autres parties du réseau continuent de fonctionner normalement. Cela limite les interruptions de travail et facilite la résolution des problèmes.

### **Avantages pour l'entreprise :**

Cette architecture apporte plusieurs bénéfices concrets :

- **Sécurité** : protection efficace contre les cyberattaques
- **Stabilité** : le système fonctionne de manière fiable
- **Productivité** : les utilisateurs peuvent travailler sans interruption
- **Économies** : moins de pannes et de temps d'arrêt
- **Conformité** : respect des règles de sécurité informatique
- **Évolutivité** : possibilité de s'adapter aux besoins futurs

Cette architecture représente une solution moderne et efficace pour sécuriser l'infrastructure informatique de l'entreprise tout en conservant une utilisation simple et productive pour tous les utilisateurs.

## Deuxième partie : Outils et méthodologie :

---

### 1. Outils utilisés :

La mise en œuvre de notre projet a nécessité l'utilisation d'un ensemble cohérent d'outils logiciels et matériels, sélectionnés pour leur pertinence dans le domaine de la cybersécurité, de la simulation réseau et de l'analyse des intrusions. Ces outils nous ont permis de construire un environnement virtuel réaliste, sécurisé et fonctionnel.

#### ➤ **GNS3 – Simulateur réseau avancé**

GNS3 (Graphical Network Simulator 3) est une solution open source nous permet de modéliser des infrastructures réseau complexes en simulant des routeurs, switches, firewalls, serveurs, etc. Grâce à GNS3, nous avons pu concevoir une architecture réseau complète, incluant une zone LAN, une DMZ, et un accès Internet, tout en y intégrant les différents équipements de sécurité. Il nous a permis également de visualiser et d'interconnecter les composants du réseau de manière intuitive.



#### ➤ **FortiGate – Pare-feu professionnel de nouvelle génération**

FortiGate est un pare-feu **NGFW** (Next Generation Firewall) très répandu dans le milieu professionnel. Il a été intégré dans notre infrastructure comme point de sécurité périmétrique.

Ses avantages par rapport à d'autres solutions open source (comme pfSense ou iptables) incluent :

- Interface de gestion intuitive via le Web GUI.
- Règles de sécurité granulaires avec inspection applicative (Layer 7).
- Support natif pour les scénarios DMZ, VPN, NAT, filtrage IP/port, etc.
- Réalisme accru : FortiGate est utilisé dans de nombreuses entreprises, ce qui donne un caractère professionnel à notre simulation.



#### ➤ **Kali Linux – Distribution spécialisée en cybersécurité**

Kali Linux est la distribution de référence pour les tests de sécurité offensive. Elle est fournie avec un large éventail d'outils de pénétration, d'analyse réseau et de forensic. Nous l'avons préférée à d'autres distributions comme Parrot OS ou BackBox pour plusieurs raisons :

- Communauté et documentation très riches.
- Compatibilité directe avec GNS3 et VMware.
- Outils préinstallés comme Nmap, Wireshark, Metasploit, permettant de simuler des attaques ou des tests d'intrusion sur notre architecture.



### ➤ **VMware Workstation – Virtualisation stable et performante**

Pour exécuter les différentes machines virtuelles utilisées dans le projet (Kali Linux, clients, serveurs, etc.), nous avons opté pour VMware Workstation. Par rapport à d'autres solutions comme VirtualBox, VMware offre :

- Meilleure performance et stabilité pour les machines virtuelles complexes.
- Intégration fluide avec GNS3, notamment via le bridge réseau.



## **2. Méthodologie adoptée**

La démarche suivie pour la réalisation du projet s'est articulée autour de quatre grandes phases :

### **a. Conception logique du réseau**

Une analyse préliminaire a permis de définir la structure globale du réseau, en identifiant les zones stratégiques nécessaires à une segmentation claire et sécurisée :

- **Zone Internet** : point d'entrée des connexions externes.
- **Zone DMZ** : héberge les services accessibles depuis l'extérieur, notamment le serveur web.
- **Zone LAN** : réservée aux clients internes et services critiques, protégée par des règles strictes.

### **b. Simulation sous GNS3**

Nous avons utilisé GNS3 pour simuler l'infrastructure complète :

- Déploiement des équipements réseau (*FortiGate, switchs L3, clients et serveurs*).
- Configuration du routage inter-VLAN via les switchs L3.
- Intégration de machines virtuelles, dont un serveur Ubuntu pour l'hébergement web.

### **c. Génération et analyse d'attaques**

Pour évaluer la robustesse du système, nous avons simulé plusieurs scénarios d'attaque à l'aide de Kali Linux, incluant :

- Scan de ports avec Nikto cible serveur web
- Injection SQL ciblant le serveur web
- XSS (Cross-Site Scripting) cible serveur web

### **d. Rapport et documentation**

L'ensemble des étapes techniques et des configurations ont été soigneusement documentées. Le rapport inclut :

- Des captures d'écran illustrant chaque phase clé.
- Une analyse critique des choix techniques.
- Une évaluation des forces et faiblesses de l'architecture mise en œuvre.

## **3. Etude des scénarios d'attaques avec EBIOS :**

### **➤ Étape 1 : Cadrage de l'étude**

#### **1. Contexte :**

Réseau d'une entreprise e-commerce, hébergeant des serveurs Web, stockage, SIEM, postes utilisateurs, zone DMZ, équipements réseau et postes administrateurs.

#### **2. Périmètre :**

Architecture représentée dans le schéma, incluant la zone DMZ, le SI interne, les postes admins et utilisateurs, les pare-feux, les serveurs critiques et les équipements réseau.

#### **3. Objectif de sécurité :**

Assurer la confidentialité, intégrité et disponibilité des données clients.

Prévenir les intrusions (ex. phishing, attaque réseau, vol de données).

Garantir la continuité des services (site web, paiement, gestion de commandes).

Protection des informations des employés et des clients sur les serveurs.  
Détection et blocage des alertes malveillantes.

## ➤ Étape 2 : Couple Source des risques et objective viser (SR / OV)

### 1. Couples SR / OV :

Source de Risque (SR)	Objectif Visé (OV)
Cybercriminels	Vol de données clients/employés sur les serveurs (Storage)
Employé malveillant	Sabotage ou fuite de données sensibles internes
Hacker externe (script kiddie)	Défiguration du site web hébergé dans la DMZ (Web)
Administrateur négligent	Mauvaise configuration du firewall, exposant la DMZ
Malware (via phishing)	Prise de contrôle de postes internes pour pivoter vers les serveurs critiques
Fournisseur tiers compromis	Entrée dans le système via une MAJ compromise ou accès VPN
Botnet/attaque DDoS	Saturation des services web publics

## ➤ Étape 3 : Étude des scénarios alternatif

### 1. Cybercriminels :

- Reconnaissance par le scan de vulnérabilités du serveur Storage
- Exploitation d'une vulnérabilité dans le service de stockage
- Escalade de privilèges pour extraction des données sensibles

### 2. Employé malveillant :

Employé = une accès et acteur légitime

### 3. Hacker externe :

- Scan pour exploitation d'une faille du CMS Web (ex. injection, XSS)
- Défiguration de site web (Defacement)

### 4. Administrateur négligent :

Exposition de ports ou services non filtrés, pouvant être exploités par un attaquant externe.

### **5. Malware (via phishing) :**

- Une campagne d'e-mails piégés entraîne le téléchargement d'un malware.
- Un accès à distance non autorisé permet à l'attaquant d'effectuer des mouvements latéraux dans le réseau, jusqu'à atteindre des serveurs critiques.

### **6. Fournisseur tiers compromis :**

- L'attaquant compromet un poste fournisseur afin de voler ses identifiants.

### **➤ Étape 4 : Étude des Scénario opérationnel**

Type de menace	Scénario opérationnel
<b>Cybercriminels</b>	<ul style="list-style-type: none"><li>• Compromission d'un compte utilisateur avec accès au stockage</li><li>• Téléchargement en masse des fichiers via une session légitime</li></ul>
<b>Employé malveillant</b>	<ul style="list-style-type: none"><li>• Employé copie des données sur clé USB ou les envoie par mail</li><li>• Suppression volontaire de fichiers sensibles ou modification de la base de données</li></ul>
<b>Hacker externe</b>	<ul style="list-style-type: none"><li>• L'administrateur oublie de faire une mise à jour, ce qui laisse une faille ouverte et permet à l'attaquant de réussir son attaque</li></ul>
<b>Administrateur négligent</b>	<ul style="list-style-type: none"><li>• Admin ouvre un port à des fins de test mais oublie de le fermer</li><li>• L'absence de restriction d'adresse IP permet un accès non autorisé</li></ul>
<b>Malware (via phishing)</b>	<ul style="list-style-type: none"><li>• L'utilisateur clique sur une pièce jointe, ce qui déclenche l'exécution d'un script malveillant</li><li>• Absence de solution EDR ou d'isolation, permettant à l'attaquant de se déplacer latéralement dans le réseau</li></ul>
<b>Fournisseur tiers compromis</b>	<ul style="list-style-type: none"><li>• Le fournisseur déploie une mise à jour infectée, introduisant un malware dans le système d'information</li><li>• Absence d'environnement de test, ce qui entraîne un déploiement direct en production</li></ul>
<b>Botnet / attaque DDoS</b>	<ul style="list-style-type: none"><li>• Aucun WAF ni protection anti-DDoS en place</li><li>• Le site n'est pas réparti en cluster ni en CDN, ce qui entraîne une indisponibilité en cas de surcharge ou de défaillance</li></ul>

## ➤ Étape 5 : Traitement du risque

Risque /Conséquences	Typologie	Impact estimé
<b>Cybercriminels :</b>	Confidentialité, Traçabilité	Très élevé

### Conséquences détaillées :

- Perte de confidentialité des données clients/employés
- Enfreinte au RGPD, obligation de notification
- Perte de confiance des utilisateurs

Risque /Conséquences	Typologie	Impact estimé
<b>Employé malveillant :</b>	Confidentialité, Intégrité	Élevé

### Conséquences détaillées :

- Atteinte à l'intégrité des données
- Risques juridiques en cas de fuite
- Temps et coût de restauration des systèmes

Risque /Conséquences	Typologie	Impact estimé
<b>Hacker externe :</b>	Intégrité, Disponibilité	Moyen à élevé

### Conséquences détaillées :

- Défiguration ou compromission du site Web
- Perte de crédibilité et d'image pour l'entreprise
- Risque de propagation de code malveillant

Risque /Conséquences	Typologie	Impact estimé
<b>Administrateur négligent :</b>	Disponibilité, Intégrité, Traçabilité	Élevé

### Conséquences détaillées :

- Ouverture d'une porte aux attaquants
- Compromission potentielle du réseau interne
- Réduction du niveau de sécurité global

Risque /Conséquences	Typologie	Impact estimé
<b>Malware (via phishing) :</b>	Intégrité, Confidentialité, Disponibilité	Très élevé

#### Conséquences détaillées :

- Prise de contrôle de postes de travail
- Accès aux serveurs critiques
- Propagation de ransomware

Risque /Conséquences	Typologie	Impact estimé
<b>Fournisseur tiers compromis :</b>	Intégrité	Très élevé

#### 4. Faisabilité du projet :

##### ➤ Ressources techniques :

- Une partie du serveur pédagogique mis à disposition par l'ENSET Mohammedia, sous l'encadrement du Pr. **Azeddine KHIAT**.
- Logiciels utilisés : GNS3, VMware, Kali Linux, FortiGate VM, Apache2, ModSecurity.
- Environnement de test hybride : travail à distance via **AnyDesk**, complété par des séances sur site.

##### ➤ Ressources matérielles :

- Accès à la Salle Info 1 en dehors des horaires de cours, sur demande officielle validée par l'administration.
- Matériel personnel des étudiants (ordinateurs portables) utilisé pour la conception et les tests.

#### 5. Organisation et mode de travail

Le travail de groupe s'est déroulé dans un cadre structuré et collaboratif, articulé autour d'une planification souple mais efficace, adaptée aux contraintes d'emploi du temps et de disponibilité des ressources.

Dès le début du projet, une coordination a été établie entre les membres du groupe afin d'assurer une progression continue et cohérente. Chaque membre a pu contribuer activement en fonction de ses compétences et de ses disponibilités, ce qui a permis une bonne répartition naturelle des tâches autour des axes suivants : architecture réseau, configuration de sécurité, et services applicatifs.



Pour faciliter l'avancement du projet et pallier les éventuelles contraintes d'accès au serveur à distance, le groupe a mis en place plusieurs modalités de travail :

- Travail à distance via la solution AnyDesk, permettant une collaboration et partage direct en temps réel sur le serveur de l'école.
- Accès sur site, rendu possible grâce à une demande officielle adressée à l'administration pour l'utilisation de la salle d'informatique 1 en dehors des horaires de cours. Cette demande, signée par notre encadrant **M. Youssefi**, a facilité la continuité du projet dans un cadre institutionnel.

**Hachani Wijdane**  
Étudiante en II-CCN  
ENSET Mohammedia  
wijdane.hachani@gmail.com – 0624920087

Madame la Secrétaire Générale  
ENSET Mohammedia

**Objet : salle info 1**

**Madame la Secrétaire Générale,**

Dans le cadre de notre projet d'innovation, nous avons besoin de travailler sur un serveur au sein de l'établissement, auquel nous devons nous connecter afin de poursuivre le développement de notre solution.


À cet effet, nous souhaiterions bénéficier de la salle **Info 1** pour les séances restantes de cette année, dans la plage horaire allant de 11h40 à 18h. Notre groupe est composé de cinq participants : **Allali Houda, Agnaou Ilyas, Sati Mariam, Hettaba Chayma et Hachani Wijdane**, et l'accès à cette salle est essentiel pour assurer la continuité de notre travail, notamment grâce à la connexion au serveur.

Cette activité s'inscrit pleinement dans la continuité de notre formation et a pour objectif de mener à bien la réalisation de notre projet d'innovation.

Dans l'attente d'une réponse favorable de votre part, nous vous remercions par avance pour l'attention portée à cette demande et restons à votre disposition pour toute information complémentaire.

Veuillez agréer, Madame la Secrétaire Générale, l'expression de nos salutations distinguées.

**Hachani Wijdane**  
Étudiante en II-CCN / Département Mathématiques et Informatique

  
M. Joussef

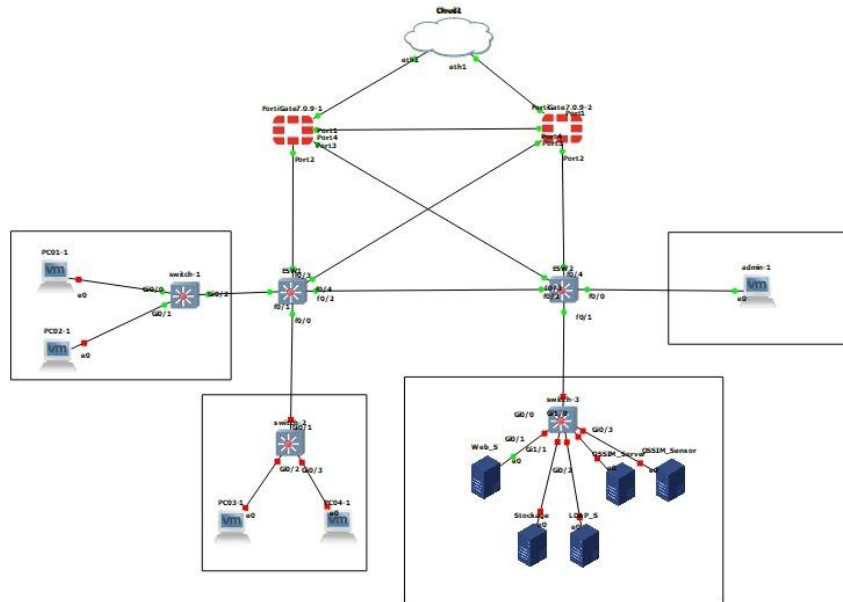


Enfin, des points réguliers d'échange ont été maintenus, que ce soit via messagerie instantanée ou lors des séances en présentiel, afin d'assurer une bonne synchronisation entre les membres et de garantir l'alignement sur les objectifs pédagogiques du projet.

## Troisième partie : Réalisation technique du projet

### 1. Architecture réseau conçue :

Cette architecture implémente une approche de sécurité en couches avec 4 zones distinctes protégées par des firewalls, suivant le principe de défense en profondeur.



L'architecture réseau comprend les éléments suivants :

- **Segment LAN** : Postes clients internes.
- **DMZ** : Serveurs web, DNS, FTP accessibles depuis l'extérieur.
- **Firewall** : Protection entre Internet, DMZ et LAN.

#### ➤ Analyse détaillée :

##### Segment LAN (Postes clients internes)

- **Fonction** : Zone de travail pour utilisateurs légitimes
- **Caractéristiques** :
  - Accès contrôlé vers Internet via firewall
  - Communication interne autorisée
- **Sécurité** : Politique restrictive, antivirus, contrôle d'accès

##### DMZ (Zone démilitarisée)

- **Services hébergés** :
  - **Serveur web** : Sites publics, applications web
  - **Serveur de stockage** : Sauvegarde de la configuration de chaque composant pour garantir la préservation des informations en cas de besoin.

- **Serveur LDAP (Lightweight Directory Access Protocol)** : Gestion centralisée des utilisateurs afin de contrôler les accès et les permissions.
- **SIEM OSSIM (Open Source Security Information Management)** : Systèmes de gestion des événements, de détection d'intrusion et de gestion des logs.
- **Principe** : Zone tampon entre Internet et LAN interne
- **Règles de filtrage** :
  - Internet → DMZ : Accès contrôlé aux services
  - DMZ → LAN : Très restrictif ou interdit
  - LAN → DMZ : Accès administratif limité

### **Firewall (Protection multicouches)**

- **Positionnement stratégique** :
  - **Périphérique** : Entre Internet et réseau interne
  - **Interne** : Entre DMZ et LAN
- **Fonctions** :
  - Filtrage de paquets (Layer 3/4)
  - Inspection applicative (Layer 7)
  - NAT/PAT pour masquage d'adresses
  - VPN pour accès distants sécurisés

## **2. Configuration des équipements réseau :**

Cette section présente la configuration des équipements réseau, incluant les switches de niveau 2 (L2), les switches de niveau 3 (L3) et le pare-feu.

### **a. Switch L3 - ESW1**

Le tableau suivant présente la configuration des interfaces du switch de niveau 3 ESW1 :

Interface	VLAN associé	Réseau	Adresse IP
F0/1	VLAN 16 SW1	192.168.16.0/24	192.168.16.5/24
F0/0	Vlan 17 SW2	192.168.17.0/24	192.168.17.5/24
F0/2	-	192.168.21.0/24	-
F0/3	-	192.168.20.0/24	

Cette configuration montre l'attribution des interfaces du switch L3 aux différents segments réseau, avec les VLAN 16 et 17 configurés respectivement sur les interfaces F0/1 et F0/0, tandis que les interfaces F0/2 et F0/3 sont connectées aux réseaux 192.168.21.0/24 et 192.168.20.0/24.

F0/1 :

Avec vlan : VLAN16\_SW1

Réseaux : 192.168.16.0/24

@ip F0/1 : 192.168.16.5/24

```
ESW1#enable
ESW1#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
ESW1(config)#ip routing
ESW1(config)#inter
ESW1(config)#interface FastEthernet0/1
ESW1(config-if)#no switchport
ESW1(config-if)#ip address 192.168.16.5 255.255.255.0
ESW1(config-if)#no shutdown
ESW1(config-if)#
*Mar 1 01:55:41.715: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up
```

F0/0 :

Avec vlan : Vlan17\_SW2

Réseaux : 192.168.17.0/24

@ip F0/0 : 192.168.17.5/24

```
ESW1#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
ESW1(config)#interface FastEthernet0/0
ESW1(config-if)#no switchport
ESW1(config-if)#ip address 192.168.17.5 255.255.255.0
ESW1(config-if)#no shutdown
ESW1(config-if)#end
ESW1#
*Mar 1 02:01:46.147: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
ESW1#
*Mar 1 02:01:47.383: %SYS-5-CONFIG_I: Configured from console by console
ESW1#write memory
Building configuration...
[OK]
```

F0/2 :

```
ESW1#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
ESW1(config)#interface FastEthernet0/2
ESW1(config-if)#no switchport
ESW1(config-if)#
*Mar 1 04:29:11.210: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/2, changed state to up
ESW1(config-if)#ip address 192.168.18.1 255.255.255.0
ESW1(config-if)#no shutdown
ESW1(config-if)#end
ESW1#
*Mar 1 04:33:26.858: %SYS-5-CONFIG_I: Configured from console by console
ESW1#write memory
Building configuration...
[OK]
ESW1#
```

F0/3

Réseau : 192.168.21.0/24

```
ESW1#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
ESW1(config)#interface FastEthernet0/3
ESW1(config-if)#no switchport
ESW1(config-if)#ip addr
*Mar 1 00:04:46.551: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/3, changed state to up
ESW1(config-if)#ip address 192.168.21.3 255.255.255.0
ESW1(config-if)#no shutdown
ESW1(config-if)#end
```

F0/3 :

Reseau : 192.168.20.0/24

```
ESW1#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
ESW1(config)#interface FastEthernet0/4
ESW1(config-if)#no switchport
ESW1(config-if)#ip address 192.168.21.3 255.255.255.0
*Mar  1 00:08:03.771: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/4, changed state to up
ESW1(config-if)#ip address 192.168.20.3 255.255.255.0
ESW1(config-if)#no shutdown
ESW1(config-if)#end
```

Routeur :

```
ESW1(config)#ip route 192.168.30.0 255.255.255.0 192.168.20.2 1
ESW1(config)#ip route 192.168.30.0 255.255.255.0 192.168.21.1 10
ESW1(config)#
```

```
ESW1#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
ESW1(config)#ip routing
ESW1(config)#ip route 192.168.29.0 255.255.255.0 192.168.20.2 1
ESW1(config)#ip route 192.168.29.0 255.255.255.0 192.168.21.1 10
```

Show routage Table :

```
ESW1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

S    192.168.29.0/24 [1/0] via 192.168.20.2
      [1/0] via 192.168.18.2
S    192.168.30.0/24 [1/0] via 192.168.20.2
      [1/0] via 192.168.18.2
C    192.168.21.0/24 is directly connected, FastEthernet0/3
C    192.168.20.0/24 is directly connected, FastEthernet0/4
C    192.168.17.0/24 is directly connected, FastEthernet0/0
C    192.168.16.0/24 is directly connected, FastEthernet0/1
C    192.168.18.0/24 is directly connected, FastEthernet0/2
ESW1#
```



### **b. Switch 1 :**

Configuration Vlan : VLAN16\_SW1

Cette configuration établit le VLAN 16 avec un nom descriptif et assigne les interfaces g0/0 et g0/1 en mode Access, permettant aux équipements connectés d'appartenir uniquement à ce VLAN sans possibilité de trunk.

```
Switch>
Switch>en
Switch#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#vlan 16
Switch(config-vlan)#name VLAN16_SW1
Switch(config-vlan)#ex
Switch(config)#inter rang
Switch(config)#inter range g0/0,g0/1
Switch(config-if-range)#switchport mode access

Switch(config-if-range)#switchport access vlan 16
Switch(config-if-range)#ex
Switch(config)#inter g0/2
Switch(config-if)#switchport mode trunk

Switch(config)#inter g0/2
Switch(config-if)#switchport trunk encapsulation dot1q
Switch(config-if)#switchport mode trunk
Switch(config-if)#end
Switch#
*Apr 28 15:03:57.441: %SYS-5-CONFIG_I: Configured from console by console
```

### **c. Switch 2 :**

Configuration Vlan17\_SW2

```
Switch#
*Apr 28 15:10:06.132: %SYS-5-CONFIG_I: Configured from console by console
Switch#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#vlan 17
Switch(config-vlan)#name Vlan17_SW2
Switch(config-vlan)#ex
Switch(config)#inter range g0/2,g0/3
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan 17
Switch(config-if-range)#ex
Switch(config)#inter G0/1

Switch(config-if)#switchport trunk encapsulation dot1q
Switch(config-if)#switchport mode trunk
Switch(config-if)#end
Switch#
*Apr 28 15:13:16.262: %SYS-5-CONFIG_I: Configured from console by console
```

#### d. Switch 3 :

```
witch>enable
witch#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
witch(config)#vlan 29
witch(config-vlan)#name VLAN29_SW3
witch(config-vlan)#ex
witch(config)#inter range g0/0,g0/1,g0/3,g0/2,g1/1
witch(config-if-range)#switchport mode access
witch(config-if-range)#switchport access vlan 29
witch(config-if-range)#ex
witch(config)#inter G1/0
witch(config-if)#switchport trunk encapsulation dot1q
witch(config-if)#switchport mode trunk
witch(config-if)#end
witch#
Jun  4 18:22:27.328: %SYS-5-CONFIG_I: Configured from console by console
```

Write memory: to save your configuration

```
witch#write memory
Building configuration...
Compressed configuration from 3823 bytes to 1707 bytes[OK]
witch#
Jun  4 18:24:22.341: %GRUB-5-CONFIG_WRITING: GRUB configuration is being updated on disk. Please wait...
Jun  4 18:24:23.048: %GRUB-5-CONFIG_WRITTEN: GRUB configuration was written to disk successfully.
```

Cette séquence confirme que la configuration du switch a été correctement sauvegardée dans la mémoire non-volatile, garantissant sa persistance après un redémarrage.

#### e. Switch L3 - ESW2

Le tableau suivant présente la configuration des interfaces du switch de niveau 3 ESW1 :

Port	Réseau	Adresse IP
F0/2	192.168.18.0/24	-
F0/0	192.168.30.0/24	-
F0/1	192.168.29.0/24	-
F0/4	192.168.22.0/24	192.168.22.3/24
F0/3	192.168.19.0/24	192.168.19.3/24

Cette configuration montre que le switch ESW2 est connecté à cinq segments réseau différents. Les interfaces F0/4 et F0/3 sont configurées avec des adresses IP spécifiques (.3 sur chaque réseau), leur permettant de fonctionner comme passerelles pour les réseaux 192.168.22.0/24 et 192.168.19.0/24, tandis que les autres interfaces (F0/2, F0/0, F0/1) sont connectées aux réseaux mais sans adresse IP assignée.



### Port F0/2 : avec Réseau 192.168.18.0/24

```
SW2#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
SW2(config)#interface FastEthernet0/2
SW2(config-if)#no switchport
SW2(config-if)#ip address 192.168.18.2 255.255.255.0
SW2(config-if)#no shutdown
SW2(config-if)#end
SW2#
Mar 1 01:10:13.291: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/2, changed state to up
SW2#write memory
Building configuration...
OK]
SW2#
```

### Port F0/0 : avec réseau 192.168.30.0/24

```
SW2#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
SW2(config)#interface FastEthernet0/0
SW2(config-if)#no switchport
SW2(config-if)#ip address 192.168.30.1 255.255.255.0
SW2(config-if)#no shutdown
SW2(config-if)#end
SW2#
Mar 1 01:14:33.911: %SYS-5-CONFIG_I: Configured from console by console
SW2#
Mar 1 01:14:35.199: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
SW2#write memory
Building configuration...
OK]
SW2#
```

### Port F0/1 : réseau 192.168.29.0/24

```
SW2#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
SW2(config)#interface FastEthernet0/1
SW2(config-if)#no switchport
SW2(config-if)#ip address 192.168.29.1 255.255.255.0
SW2(config-if)#no shutdown
SW2(config-if)#end
SW2#
Mar 1 01:19:33.691: %SYS-5-CONFIG_I: Configured from console by console
SW2#
Mar 1 01:19:34.771: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up
SW2#write memory
Building configuration...
OK]
SW2#
```

### Port 0/4 : Réseau 192.168.22.3/24

```
ESW2#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
ESW2(config)#interface FastEthernet0/4
ESW2(config-if)#no switchport
ESW2(config-if)#ip address 192.168.22.3 255.255.255.0
ESW2(config-if)#no shutdown
ESW2(config-if)#end
```

### Port 0/3 : Réseau 192.168.19.3/24

```
ESW2#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
ESW2(config)#interface FastEthernet0/3
ESW2(config-if)#no switchport
ESW2(config-if)#ip address 192.168.22.3 255.255.255.0
*Mar 1 00:08:20.291: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/3, changed state to up
ESW2(config-if)#ip address 192.168.19.3 255.255.255.0
ESW2(config-if)#no shutdown
ESW2(config-if)#end
```

### Routeur :

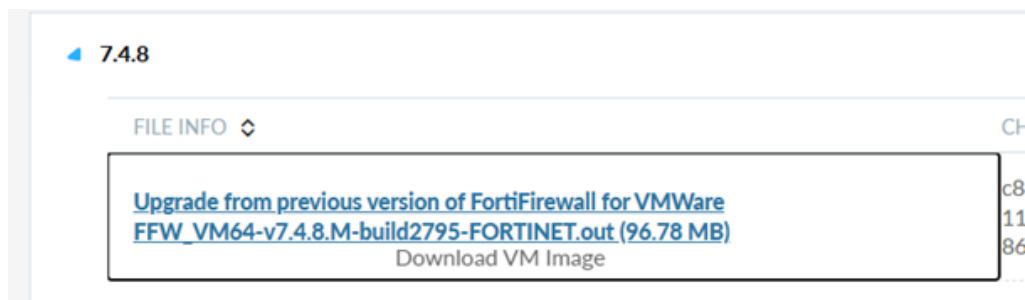
```
ESW2#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
ESW2(config)#ip route 192.168.17.0 255.255.255.0 192.168.22.2 1
ESW2(config)#ip route 192.168.17.0 255.255.255.0 192.168.19.1 10
ESW2(config)#ip route 192.168.17.0 255.255.255.0 192.168.22.2 1
ESW2(config)#ip route 192.168.17.0 255.255.255.0 192.168.19.1 10
ESW2(config)#
```

### f. Configuration des Firewalls – FortiGate :

- **FortiGate 1** : Ports 1 à 4 configurés pour la segmentation du réseau.

Le firewall FortiGate 1 utilise une configuration multi-ports pour assurer la segmentation réseau :

**Ports 1 à 4** : Chaque port est dédié à un segment réseau spécifique, permettant une séparation physique et logique des zones de sécurité. Cette approche facilite l'application de politiques de sécurité granulaires entre les différents segments.



### Port 1 :

```
FortiGate-VM64-KVM # config system interface
FortiGate-VM64-KVM (interface) # edit port1
FortiGate-VM64-KVM (port1) # set ip 10.10.7.230 255.255.252.0
FortiGate-VM64-KVM (port1) # set allowaccess ping https ssh http
```

Port 2 :

```
FortiGate-VM64-KVM (interface) # edit port2
FortiGate-VM64-KVM (port2) # set ip 192.168.21.1 255.255.255.0
FortiGate-VM64-KVM (port2) # set allowaccess ping https ssh http
```

Port 3 :

```
FortiGate-VM64-KVM (interface) # edit port3
FortiGate-VM64-KVM (port3) # set ip 192.168.19.1 255.255.255.0
FortiGate-VM64-KVM (port3) # set allowaccess ping https ssh http
```

Port 4 :

```
FortiGate-VM64-KVM (interface) # edit port4
FortiGate-VM64-KVM (port4) # set ip 192.168.23.1 255.255.255.0
FortiGate-VM64-KVM (port4) # set allowaccess ping https ssh http
```

Configuration de Routage :

```
FortiGate-VM64-KVM # config router static
FortiGate-VM64-KVM (static) # edit 1
new entry '1' added

FortiGate-VM64-KVM (1) # set dst 192.168.16.0 255.255.255.0
FortiGate-VM64-KVM (1) # set gateway 192.168.21.3
command parse error before 'gateway'
Command fail. Return code -61

FortiGate-VM64-KVM (1) # set gateway 192.168.21.3
FortiGate-VM64-KVM (1) # set device port2
```

```
FortiGate-VM64-KVM (static) # edit 2
new entry '2' added

FortiGate-VM64-KVM (2) # set dst 192.168.17.0 255.255.255.0
FortiGate-VM64-KVM (2) # set gateway 192.168.21.3
FortiGate-VM64-KVM (2) # set device port2
```

```

FortiGate-VM64-KVM (static) # edit 3
new entry '3' added

FortiGate-VM64-KVM (3) # set dst 192.168.29.0 255.255.255.0

FortiGate-VM64-KVM (3) # set gateway 192.168.19.3

FortiGate-VM64-KVM (3) # set device port3

FortiGate-VM64-KVM (3) # next

```

```

FortiGate-VM64-KVM (static) # edit 4
new entry '4' added

FortiGate-VM64-KVM (4) # set dst 192.168.30.0 255.255.255.0

FortiGate-VM64-KVM (4) # set gateway 192.168.19.3

FortiGate-VM64-KVM (4) # set device port3

FortiGate-VM64-KVM (4) # next

```

Cette configuration permet au FortiGate 1 de fonctionner comme un point de contrôle central pour tous les échanges entre les segments réseau, garantissant une sécurité optimale de l'infrastructure.

- **FortiGate 2** : Ports 2, 3, 4 connectés à divers segments internes.

Le firewall FortiGate 2 est configuré avec une **architecture tri-ports** pour la gestion des segments internes :

**Ports 2, 3, 4** : Ces trois interfaces sont dédiées à la connexion et au contrôle de différents segments du réseau interne, permettant une segmentation fine des zones de travail et des services.

Port 2 :

```

FortiGate-FW # config system interface

FortiGate-FW (interface) # edit port2

FortiGate-FW (port2) # set mode static

FortiGate-FW (port2) # set ip 192.168.22.2 255.255.255.0

FortiGate-FW (port2) # set allowaccess ping http https ssh

```

Port 3 :

```
FortiGate-FW (interface) # edit port3
FortiGate-FW (port3) # set mode static
FortiGate-FW (port3) # set ip 192.168.20.2 255.255.255.0
FortiGate-FW (port3) # set allowaccess ping http https ssh
```

Routage :

Configuration du routage avec le réseau 192.168.16.0/24.

```
FortiGate-FW # config router static
FortiGate-FW (static) # edit 1
new entry '1' added
FortiGate-FW (1) # set dst 192.168.16.0 255.255.255.0
FortiGate-FW (1) # set gateway 192.168.20.3
FortiGate-FW (1) # set device port3
```

Configuration du routage avec le réseau 192.168.17.0/24.

```
FortiGate-FW (static) # edit 2
new entry '2' added
FortiGate-FW (2) # set dst 192.168.17.0 255.255.255.0
FortiGate-FW (2) # set gateway 192.168.20.3
FortiGate-FW (2) # set device port3
```

Configuration du routage avec le réseau 192.168.29.0/24.

```
FortiGate-FW (static) # edit 3
FortiGate-FW (3) # set dst 192.168.29.0 255.255.255.0
FortiGate-FW (3) # set gateway 192.168.22.3
FortiGate-FW (3) # set device port2
```

Configuration du routage avec le réseau 192.168.30.0/24.

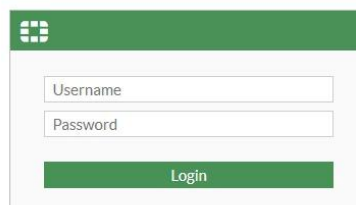
```
FortiGate-FW (static) # edit 4
new entry '4' added
FortiGate-FW (4) # set dst 192.168.30.0 255.255.255.0
FortiGate-FW (4) # set gateway 192.168.22.3
FortiGate-FW (4) # set device port2
```

Cette approche permet de vérifier le principe de haute disponibilité (HA). Le pare-feu 2 peut remplacer le pare-feu 1 en cas de panne ou lors de la mise à jour de ce dernier.

## ➤ L'interface d'administration de pare-feu FortiGate :

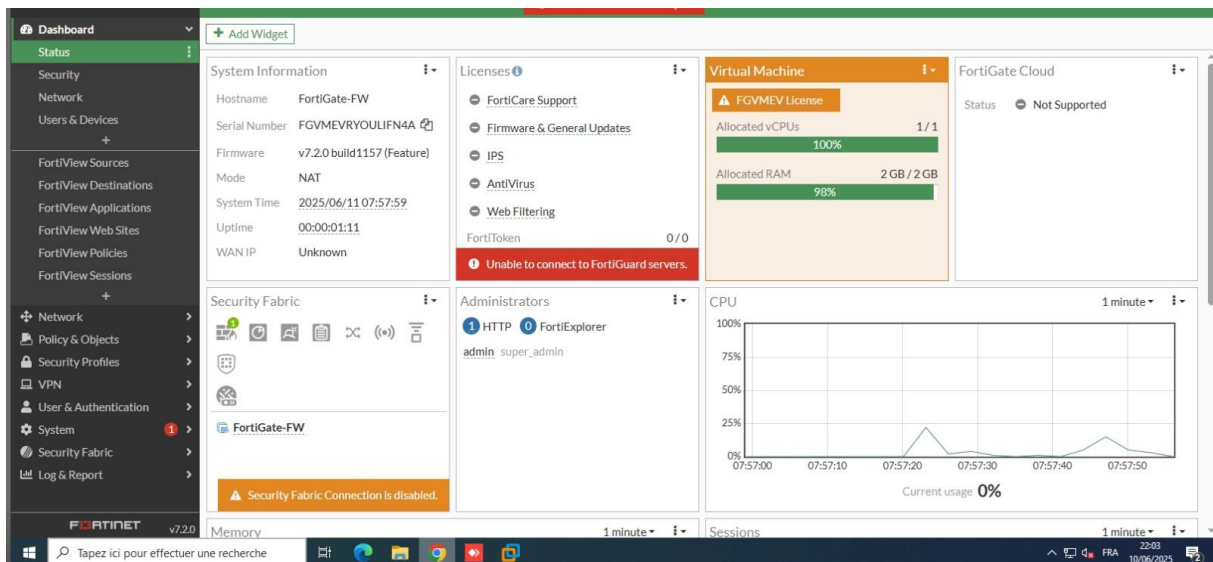
### 1. Page de connexion :

Interface de login simple avec champs username/password, cohérente avec l'identité visuelle Fortinet.



The image shows a simple login form for FortiGate. It has a green header with the Fortinet logo. Below the header, there are two input fields: 'Username' and 'Password'. At the bottom, there is a green 'Login' button.

### 2. Tableau de bord principal :



Cette figure montre le tableau de bord (Dashboard) du FortiGate-FW avec plusieurs widgets d'information :

- **Informations système :** Hostname, numéro de série, firmware v7.2.0, mode NAT



- **Licences** : État des licences FortiCare, mises à jour firmware, IPS, antivirus, filtrage web
- **Machine virtuelle** : Licence FGVMEV, allocation CPU (100%) et RAM (98%)
- **FortiGate Cloud** : Statut "Non supporté"
- **Security Fabric** : Connexion désactivée avec avertissement
- **Graphiques** : Utilisation CPU en temps réel et autres métriques système.

### 3. Configuration des interfaces réseau :

Name	Type	Members	IP/Netmask	Administrative Access	DHCP Clients	DHCP Ranges	Ref.
<b>802.3ad Aggregate</b>							
fortilink	802.3ad Aggregate		Dedicated to FortiSwitch	PING Security Fabric Connection		10.255.1.2-10.255.1.254	2
<b>Physical Interface</b>							
config port2 (port2)	Physical Interface		192.168.22.129/255.255.255.0	PING HTTPS SSH HTTP			4
config port3 (port3)	Physical Interface		192.168.20.130/255.255.255.0	PING HTTPS SSH HTTP			4
port1	Physical Interface		192.168.142.145/255.255.255.0	PING HTTPS SSH HTTP			0
<b>Tunnel Interface</b>							

Cette page présente la configuration des interfaces réseau :

- **Interfaces physiques** : port1, port2 (config port2), port3 (config port3) avec leurs adresses IP respectives
- **Agrégation 802.3ad** : Interface "fortilink" dédiée au FortiSwitch
- **Services autorisés** : PING, HTTPS, SSH, HTTP selon les interfaces
- **Plages DHCP** : Configuration des pools d'adresses IP

### 4. Politiques de pare-feu :

Name	Source	Destination	Schedule	Service	Action	NAT	Security Profiles	Log	Bytes
<b>config port2 (port2) → config port3 (port3)</b>									
Allow_Admin	admin	D16	always	ALL	ACCEPT	Enabled	no-inspection	UTM	0 B
<b>config port3 (port3) → config port2 (port2)</b>									
C_Employer	D16	admin	always	ALL	ACCEPT	Enabled	no-inspection	UTM	0 B
<b>Implicit</b>									
Implicit Deny	all	all	always	ALL	DENY			Disabled	0 B

Cette section montre les règles de pare-feu configurées :

- **Règles inter-interfaces** : Communication entre port2 et port3
- **Règle Allow\_Admin** : Accès administrateur depuis admin vers D16/D17
- **Règle C\_Employer** : Trafic depuis D16 vers admin et serveur web
- **Règle implicite de déni** : Politique par défaut qui bloque tout trafic non autorisé (DENY/Disabled)

### **3. Configuration des Serveurs :**

#### **a. Serveur LDAP :**

Un serveur LDAP a été configuré pour gérer l'authentification centralisée des utilisateurs au sein du réseau simulé.

#### **Vérification du service slapd**

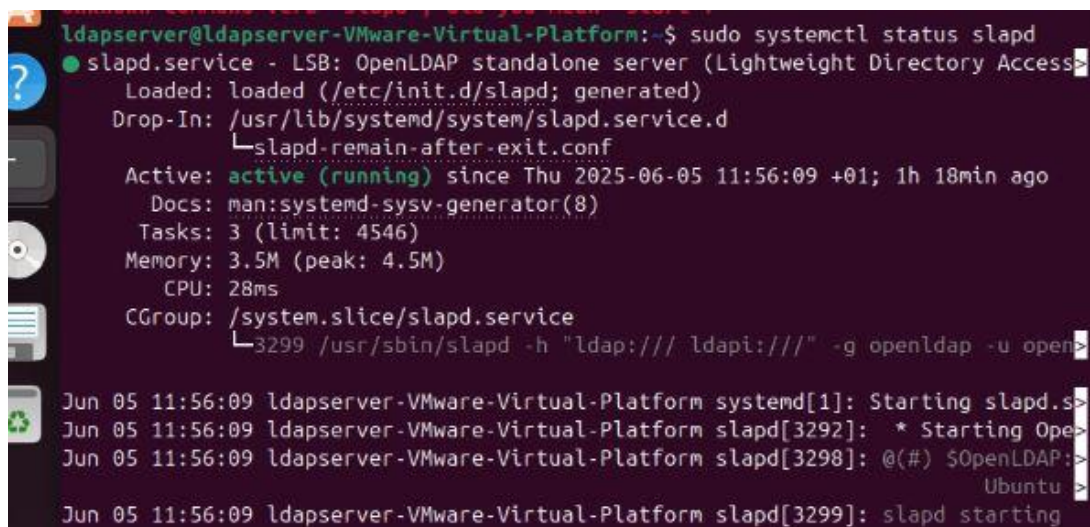
La commande suivante permet de vérifier si le service LDAP est actif :

\$ sudo systemctl status slapd

Résultat attendu :

slapd.service - LSB: OpenLDAP standalone server

Active: active (running)



```
ldapservers@ldapservers-VMware-Virtual-Platform:~$ sudo systemctl status slapd
● slapd.service - LSB: OpenLDAP standalone server (Lightweight Directory Access>
   Loaded: loaded (/etc/init.d/slapd; generated)
   Drop-In: /usr/lib/systemd/system/slapd.service.d
            └─slapd-remain-after-exit.conf
   Active: active (running) since Thu 2025-06-05 11:56:09 +01; 1h 18min ago
     Docs: man:systemd-sysv-generator(8)
    Tasks: 3 (limit: 4546)
   Memory: 3.5M (peak: 4.5M)
      CPU: 28ms
   CGroup: /system.slice/slapd.service
            └─3299 /usr/sbin/slapd -h "ldap:/// ldapi:///" -g openldap -u open>

Jun 05 11:56:09 ldapservers-VMware-Virtual-Platform systemd[1]: Starting slapd.s>
Jun 05 11:56:09 ldapservers-VMware-Virtual-Platform slapd[3292]: * Starting Ope>
Jun 05 11:56:09 ldapservers-VMware-Virtual-Platform slapd[3298]: @(#) $OpenLDAP:>
                                                    Ubuntu >
Jun 05 11:56:09 ldapservers-VMware-Virtual-Platform slapd[3299]: slapd starting
```

- **Ajout d'utilisateurs via LDIF**

Un fichier **users.ldif** contenant plusieurs utilisateurs a été créé :

dn: uid=user1,ou=people,dc=entreprise,dc=ccn

objectClass: inetOrgPerson

cn: User One

sn: One

givenName: User

sudo ldapadd -x -D "cn=admin,dc=entreprise,dc=ccn" -W -f users.ldif \

-H ldap://192.168.29.5



```

ldapserver@ldapserver-VMware-Virtual-Platform:~$ ldapsearch -x -H ldap://localhost -b dc=entreprise,dc=ccn

# extended LDIF
#
# LDAPv3
# base <dc=entreprise,dc=ccn> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#
# entreprise.ccn
dn: dc=entreprise,dc=ccn
objectClass: top
objectClass: dcObject
objectClass: organization
o: Entreprise
dc: entreprise

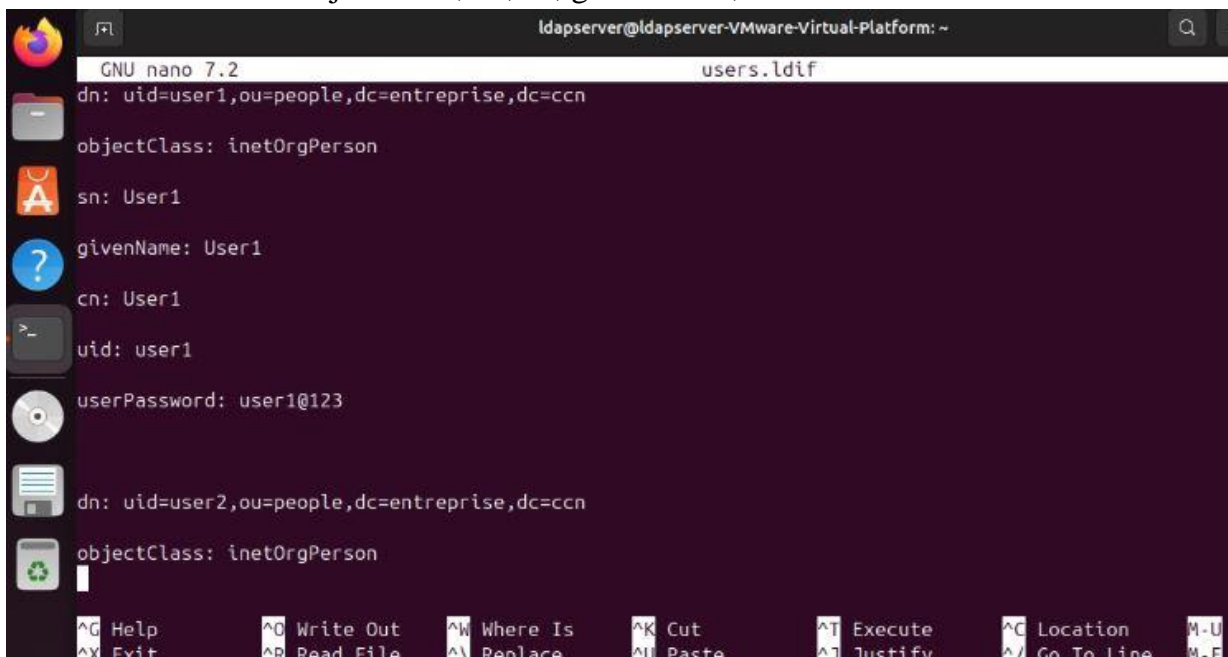
# search result
search: 2
result: 0 Success

# numResponses: 2
..

ldapserver@ldapserver-VMware-Virtual-Platform:~$ ldapwhoami -x -H ldap://192.168.77.5 -D "cn=admin,dc=entreprise,dc=ccn" -W
Enter LDAP Password:
dn:cn=admin,dc=entreprise,dc=ccn
ldapserver@ldapserver-VMware-Virtual-Platform:~$ nano users.ldif

```

Le fichier users.ldif dans l'éditeur nano avec la définition du premier utilisateur (User1) incluant les attributs objectClass, cn, sn, givenName, uid et userPassword.



```

GNU nano 7.2 users.ldif
dn: uid=user1,ou=people,dc=entreprise,dc=ccn
objectClass: inetOrgPerson
sn: User1
givenName: User1
cn: User1
uid: user1
userPassword: user1@123
dn: uid=user2,ou=people,dc=entreprise,dc=ccn
objectClass: inetOrgPerson

```

```
GNU nano 7.2 users.ldif *
dn: uid=user3,ou=people,dc=entreprise,dc=ccn
objectClass: inetOrgPerson
objectClass: top
cn: User Three
sn: Three
givenName: User
uid: user3
mail: user3@entreprise.ccn
userPassword: 123

dn: uid=user4,ou=people,dc=entreprise,dc=ccn
objectClass: inetOrgPerson
objectClass: top
cn: User Four
sn: Four
givenName: User
uid: user4
mail: user4@entreprise.ccn
userPassword: 123

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace  ^U Paste     ^J Justify   ^_ Go To Line
```

L'exécution de la commande ldapadd pour ajouter les utilisateurs du fichier LDIF à l'annuaire LDAP, confirmant l'ajout réussi des entrées user3 et user4 :

```
ldapserver@ldapserver-VMware-Virtual-Platform:~$ sudo nano users.ldif
ldapserver@ldapserver-VMware-Virtual-Platform:~$ ldapadd -x -D "cn=admin,dc=entreprise,dc=ccn" -W -f users.ldif -H ldap://192.168.77.5
Enter LDAP Password:
adding new entry "uid=user3,ou=people,dc=entreprise,dc=ccn"

adding new entry "uid=user4,ou=people,dc=entreprise,dc=ccn"

ldapserver@ldapserver-VMware-Virtual-Platform:~$
```

- **Authentification des utilisateurs**

Test d'authentification admin :

```
ldapwhoami -x -H ldap://192.168.29.5 \
-D "cn=admin,dc=entreprise,dc=ccn" -W
```

Test utilisateur standard :

```
ldapwhoami -x -H ldap://192.168.29.5 \
-D "uid=user1,ou=people,dc=entreprise,dc=ccn" -w abc123
```

Voilà l'écran de connexion Ubuntu du poste client PC01, démontrant l'intégration de l'authentification LDAP au niveau du système :



Le test de connectivité réseau (ping 192.168.29.5) depuis le poste client vers le serveur LDAP, confirmant la communication réseau :

```
pc01@pc01-VMware-Virtual-Platform:~$ ping 192.168.29.5
PING 192.168.29.5 (192.168.29.5) 56(84) bytes of data.
64 bytes from 192.168.29.5: icmp_seq=1 ttl=64 time=0.656 ms
64 bytes from 192.168.29.5: icmp_seq=1 ttl=64 time=11.1 ms (DUP!)
64 bytes from 192.168.29.5: icmp_seq=2 ttl=64 time=0.335 ms
64 bytes from 192.168.29.5: icmp_seq=2 ttl=64 time=5.62 ms (DUP!)
64 bytes from 192.168.29.5: icmp_seq=2 ttl=64 time=7.17 ms (DUP!)
64 bytes from 192.168.29.5: icmp_seq=2 ttl=64 time=15.1 ms (DUP!)
64 bytes from 192.168.29.5: icmp_seq=3 ttl=64 time=0.473 ms
64 bytes from 192.168.29.5: icmp_seq=3 ttl=64 time=6.46 ms (DUP!)
64 bytes from 192.168.29.5: icmp_seq=3 ttl=64 time=7.95 ms (DUP!)
64 bytes from 192.168.29.5: icmp_seq=3 ttl=64 time=12.4 ms (DUP!)
^C
--- 192.168.29.5 ping statistics ---
3 packets transmitted, 3 received, +7 duplicates, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 0.335/6.733/15.093/4.917 ms
```

Le test de connectivité réseau (ping 192.168.29.5) depuis le poste client vers le serveur LDAP, confirmant la communication réseau :

```
pc01@pc01-VMware-Virtual-Platform:~$  
pc01@pc01-VMware-Virtual-Platform:~$ ldapwhoami -x -D "uid=user1,ou=people,dc=entreprise,dc=ccn" -w abc123 -H ldap://192.168.29.5  
dn:uid=user1,ou=people,dc=entreprise,dc=ccn  
pc01@pc01-VMware-Virtual-Platform:~$
```

**b. Serveur OSSIM AlientVault (Stratégies de détection des intrusions) :**

- **Détection par signature** : Basée sur des motifs connus d'attaques.
- **Détection par anomalie** : Comparaison du trafic à un comportement normal établi.
- **Détection comportementale** : Analyse fine des flux TCP/IP pour identifier des comportements atypiques.

**Centralisation des Logs avec AlienVault OSSIM**

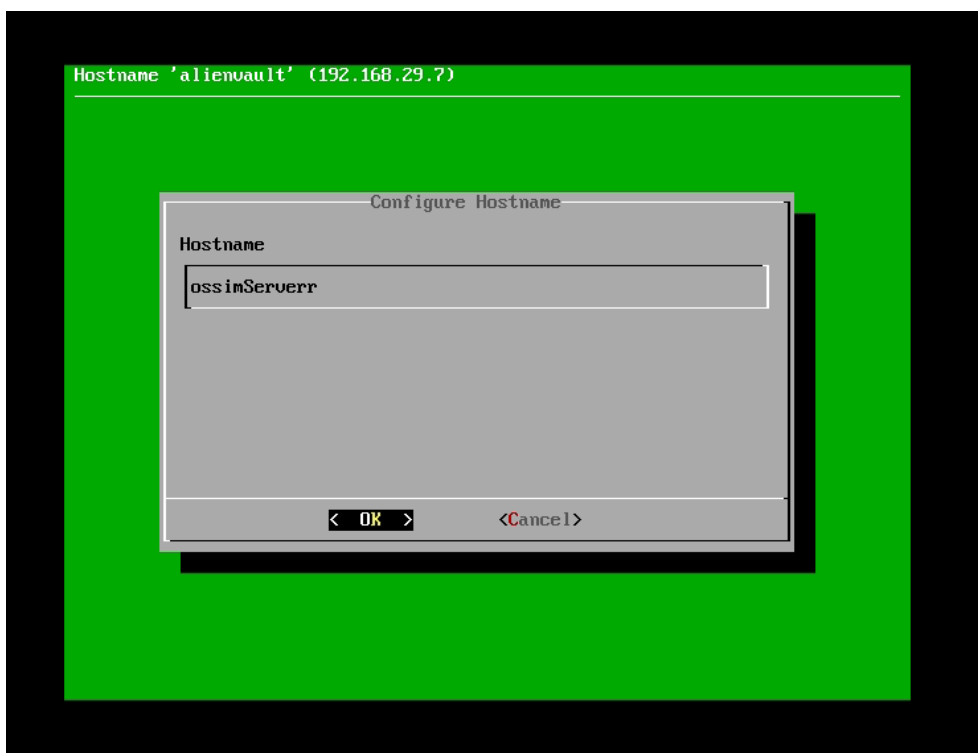
Afin de centraliser et d'analyser les journaux de sécurité générés par les différents composants du réseau (pare-feu FortiGate, serveur web Apache), nous avons déployé un système de gestion des informations et des événements de sécurité (SIEM) en utilisant AlienVault OSSIM.



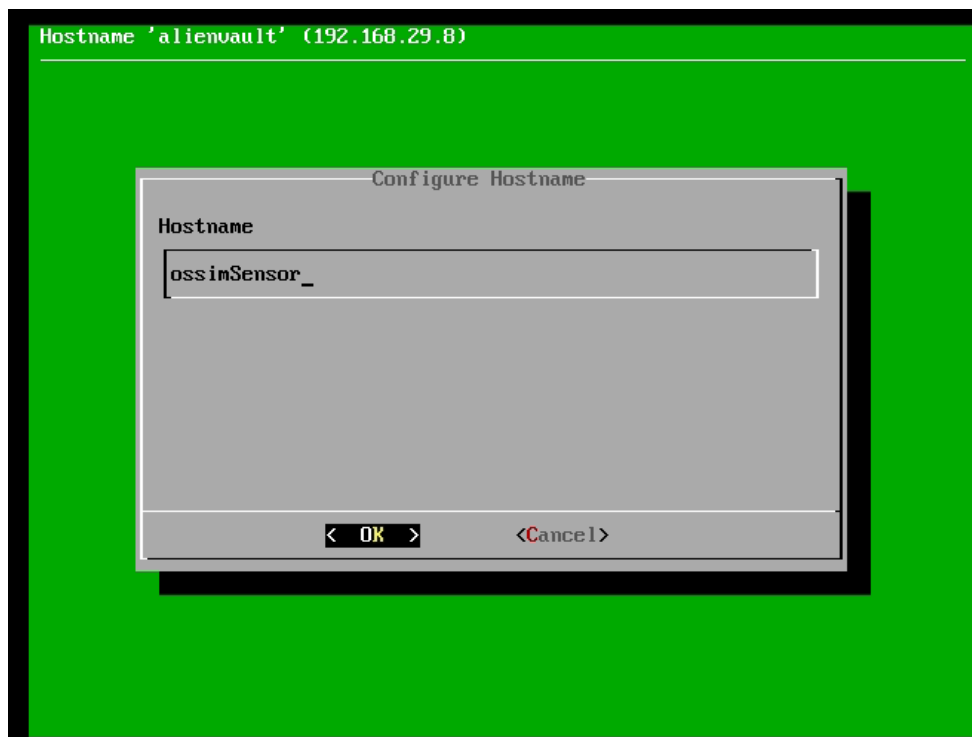
The screenshot shows the 'Configure the network' section of the AlienVault OSSIM web interface. At the top, there is a logo with three icons: a skull, a 'V', and an elephant, with the text 'ALIEN VAULT OSSIM' below it. The main heading is 'Configure the network'. Below this, a text box explains the gateway: 'The gateway is an IP address (four numbers separated by periods) that indicates the gateway router, also known as the default router. All traffic that goes outside your LAN (for instance, to the internet) is sent through this router. In rare circumstances, you may have no router; in that case, you can leave this blank. If you don't know the proper answer to this question, consult your network administrator.' Below the text box, there is a label 'Gateway:' followed by a text input field containing the IP address '192.168.1.1'. At the bottom of the form, there are three buttons: 'Screenshot', 'Go Back', and 'Continue'.



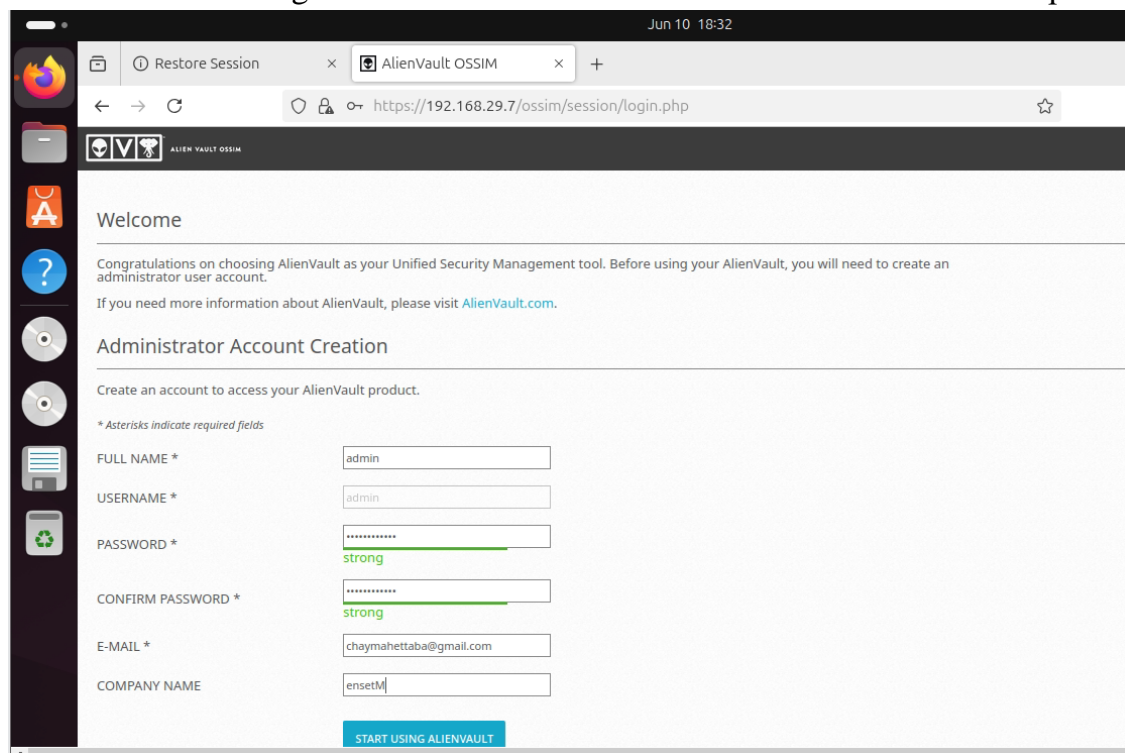
Configuration de ossim Server :  
Modification du nom



## Configuration de OSSIM sensor: Modification du nom

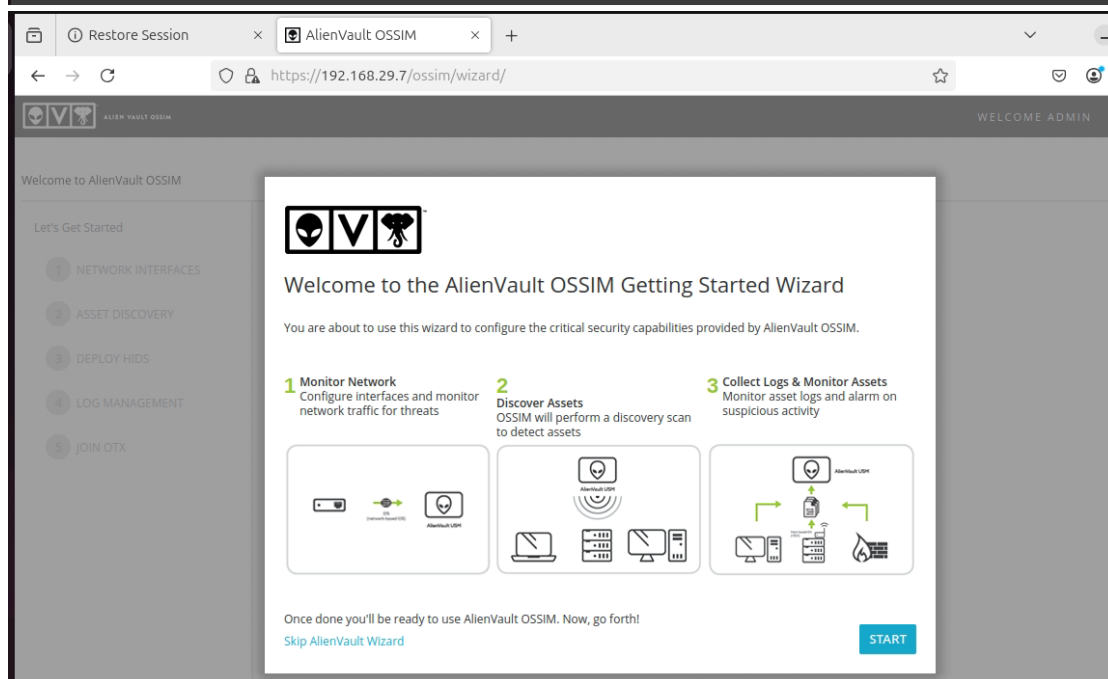
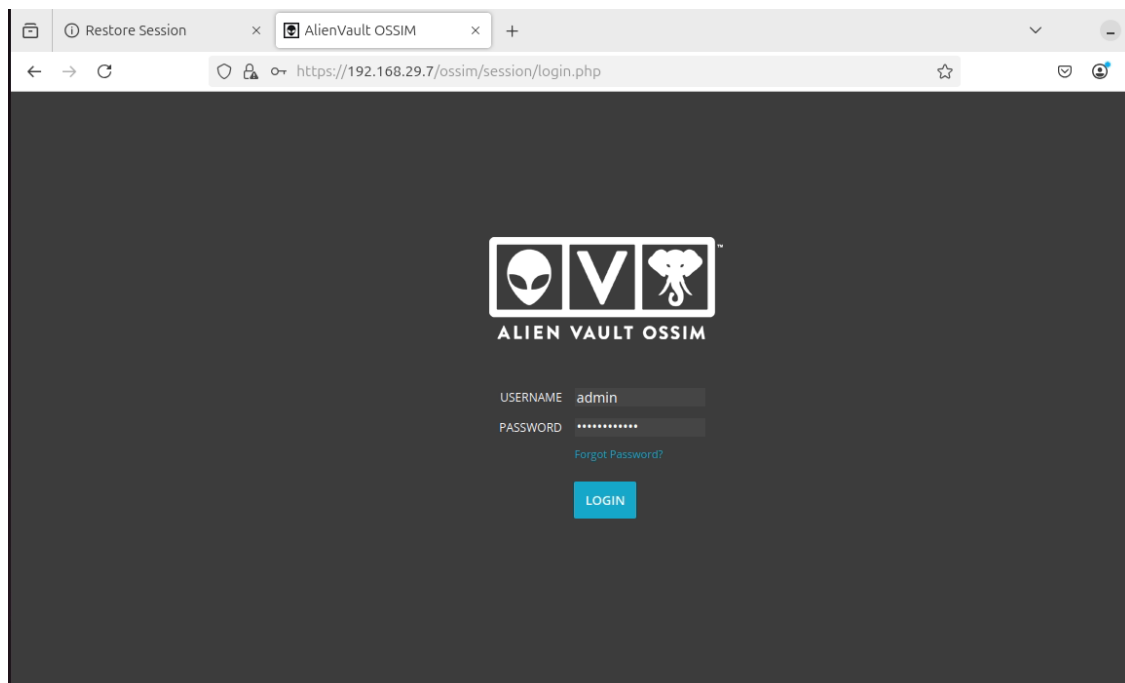


Connexion à notre OSSIM depuis la machine d'administration.  
Commencer la configuration en saisissant le nom d'utilisateur et le mot de passe.

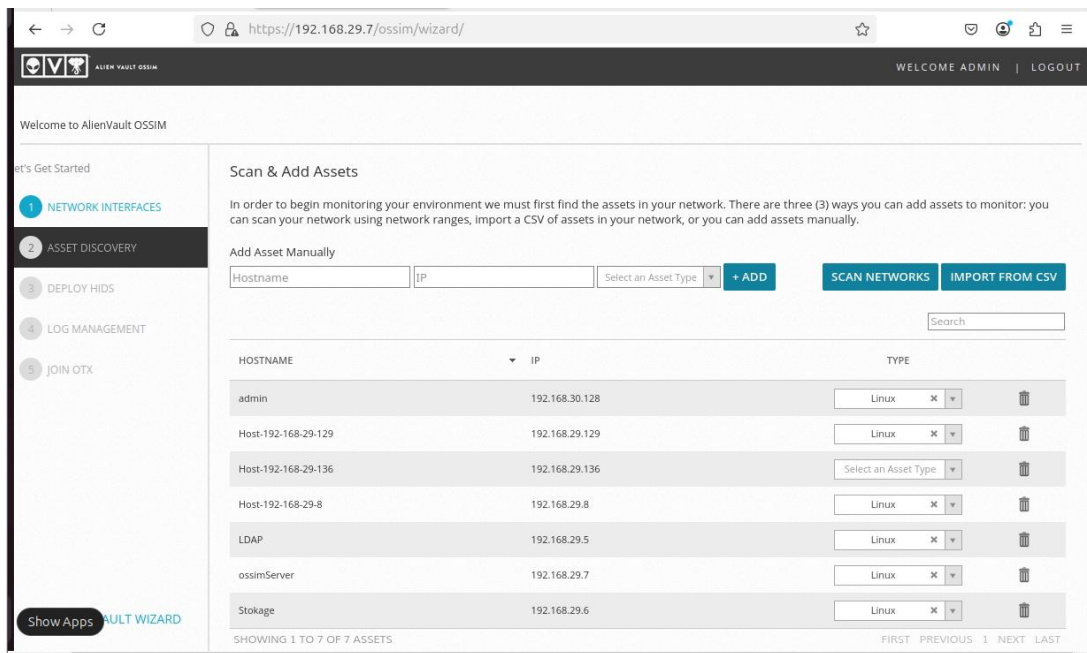




Connexion à notre SIEM.

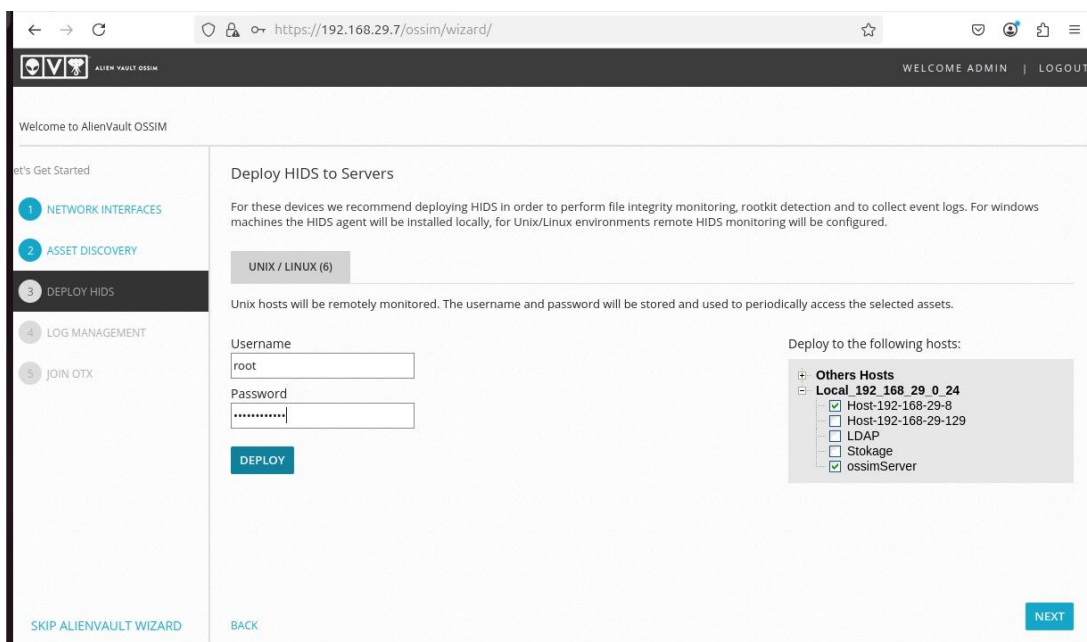


Ajout des serveurs concernés pour la gestion des logs et la sécurité : serveur web, serveur LDAP, serveur de stockage, et compte administrateur.

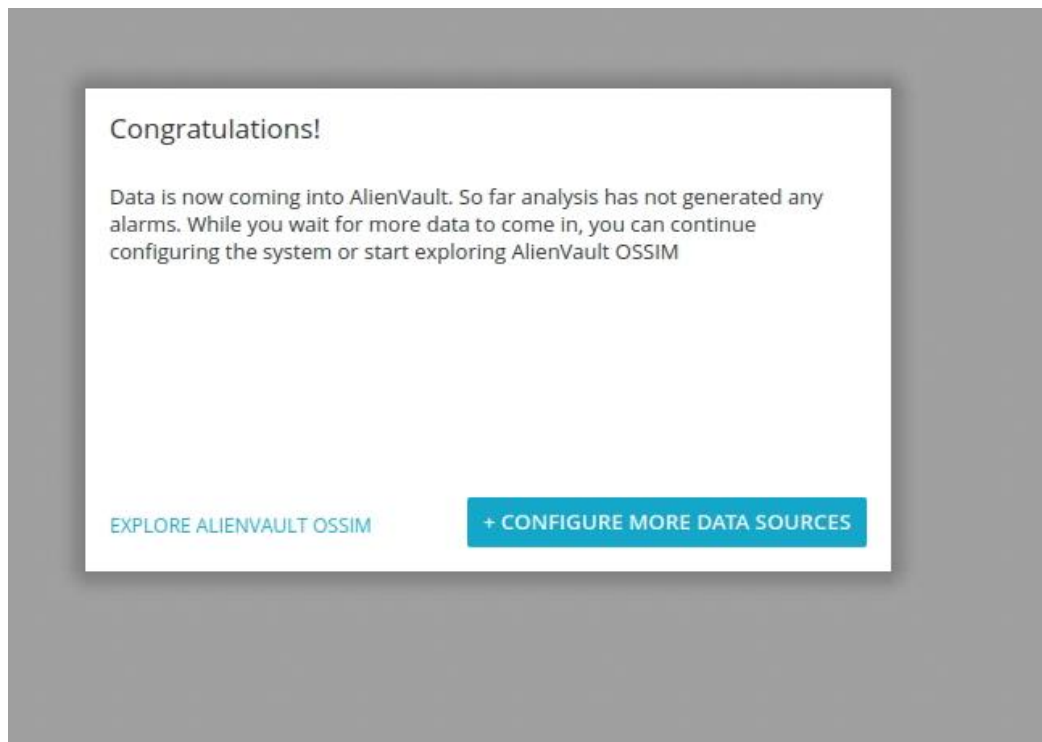


Déploiement des agents OSSIM sur les serveurs.

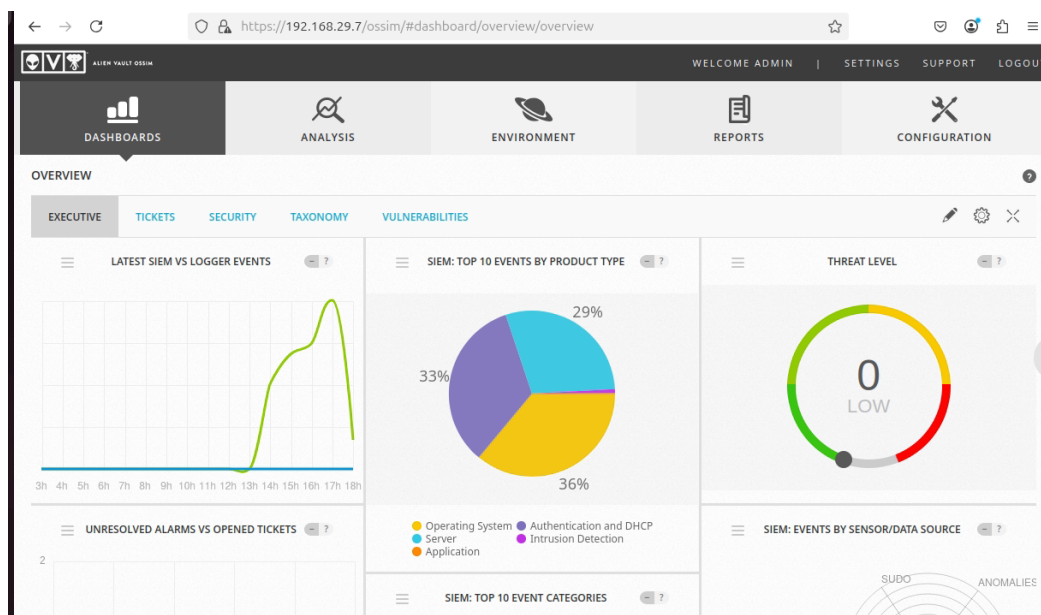
Pour déployer tous les serveurs dans notre SIEM, il suffit de cliquer sur l'adresse IP correspondant à chaque serveur et de saisir les informations de connexion pour chaque serveur de la même manière.







Dashboard /le tableau de bord de notre SIEM:



Configuration de Suricata sur le capteur (Sensor).

Accès au dossier Suricata

```
ossimSensor: #  
ossimSensor:~# cd /etc/suricata  
ossimSensor:/etc/suricata#
```

Configuration du fichier suricata.yaml

```
ossimSensor:/etc/suricata# nano suricata.yaml
```

En modifiant EXTERNAL\_NET à any, tout le trafic provenant de n'importe quelle adresse IP est considéré comme externe :

```
GNU nano 2.2.4      File: suricata.yaml      Modified

default-rule-path: /etc/snort/rules
classification-file: /etc/suricata/classification.config
reference-config-file: /etc/suricata/reference.config

# Holds variables that would be used by the engine.
vars:
# Holds the address group vars that would be passed in a Signature.
# These would be retrieved during the Signature address parsing stage.
address-groups:
  HOME_NET: "[192.168.0.0/16,172.16.0.0/12,10.0.0.0/8]"
  EXTERNAL_NET: any
  HTTP_SERVERS: "$HOME_NET"
  SMTP_SERVERS: "$HOME_NET"
  SQL_SERVERS: "$HOME_NET"
  DNS_SERVERS: "$HOME_NET"
  TELNET_SERVERS: "$HOME_NET"
  AIM_SERVERS: "$EXTERNAL_NET"
  DNP3_SERVER: "$HOME_NET"
  DNP3_CLIENT: "$HOME_NET"
  MODBUS_CLIENT: "$HOME_NET"
  MODBUS_SERVER: "$HOME_NET"
  ENIP_CLIENT: "$HOME_NET"
  ENIP_SERVER: "$HOME_NET"

# Holds the port group vars that would be passed in a Signature.

^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is   ^V Next Page  ^U UnCut Text ^T To Spell
```

Redémarrer Suricata.

```
ossimSensor:/etc/suricata# service suricata restart
Stopping suricata:  done.
Starting suricata in IDS (pcap) mode... done.
ossimSensor:/etc/suricata# _
```

### Rôle dans l'architecture globale :

Le système AlienVault OSSIM joue un rôle essentiel dans notre architecture sécurisée :

- Il **centralise tous les logs** du réseau.
- Il permet une **corrélation intelligente** des événements.
- Il offre une **vue consolidée** des alertes provenant de ModSecurity, FortiGate, etc.
- Il facilite la **détection proactive des intrusions** et la réponse aux incidents.

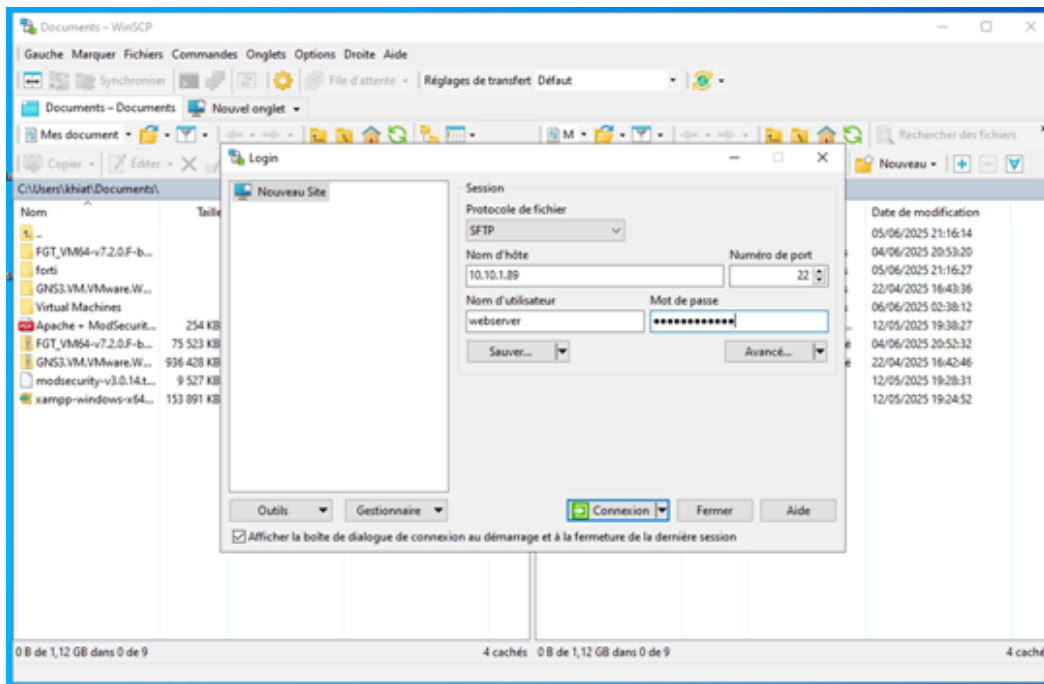
### c. Configuration du serveur Web :

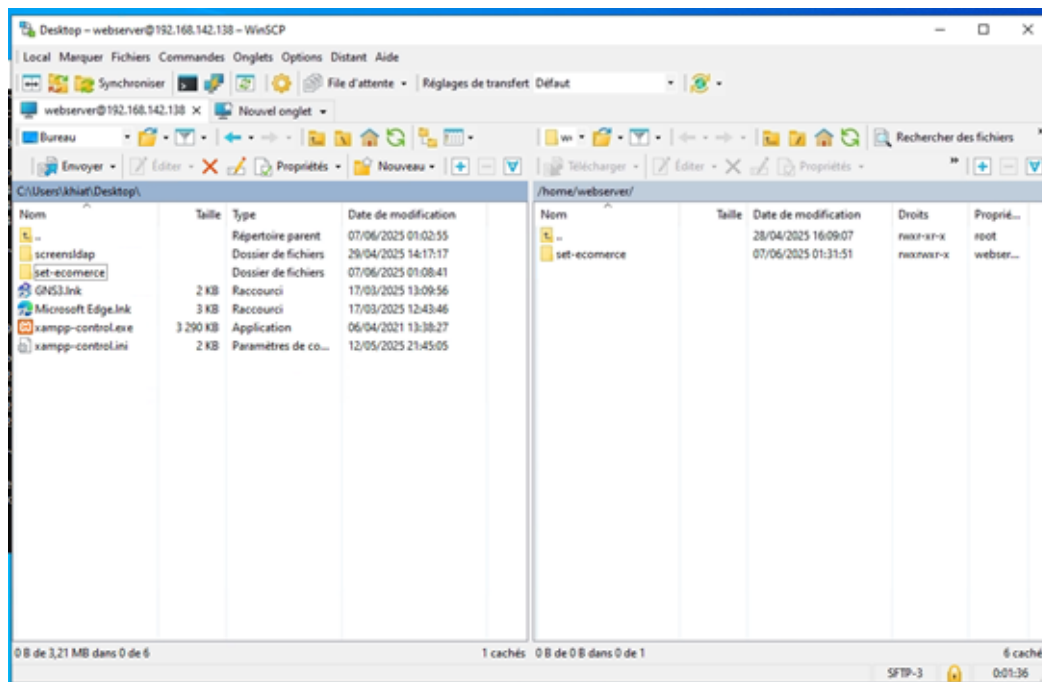
#### Installation d'Apache2 :

\$ sudo apt update && sudo apt install apache2

```
webserver@webserver:~$ sudo apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
```

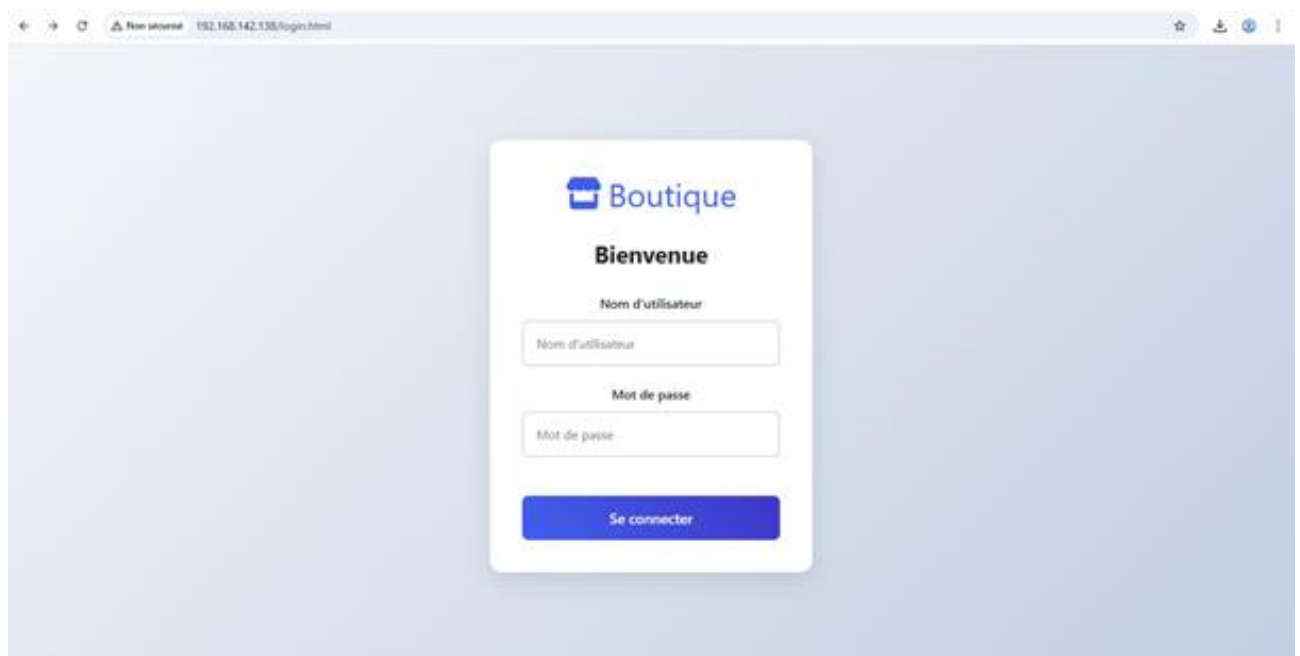
L'intégration d'un site web personnalisé dans le serveur Apache2 permet de déployer du contenu web spécifique dans l'environnement de la DMZ. Cette étape consiste à transférer les fichiers du site web depuis un poste de développement vers le serveur web et à configurer Apache pour servir ce contenu de manière sécurisée.



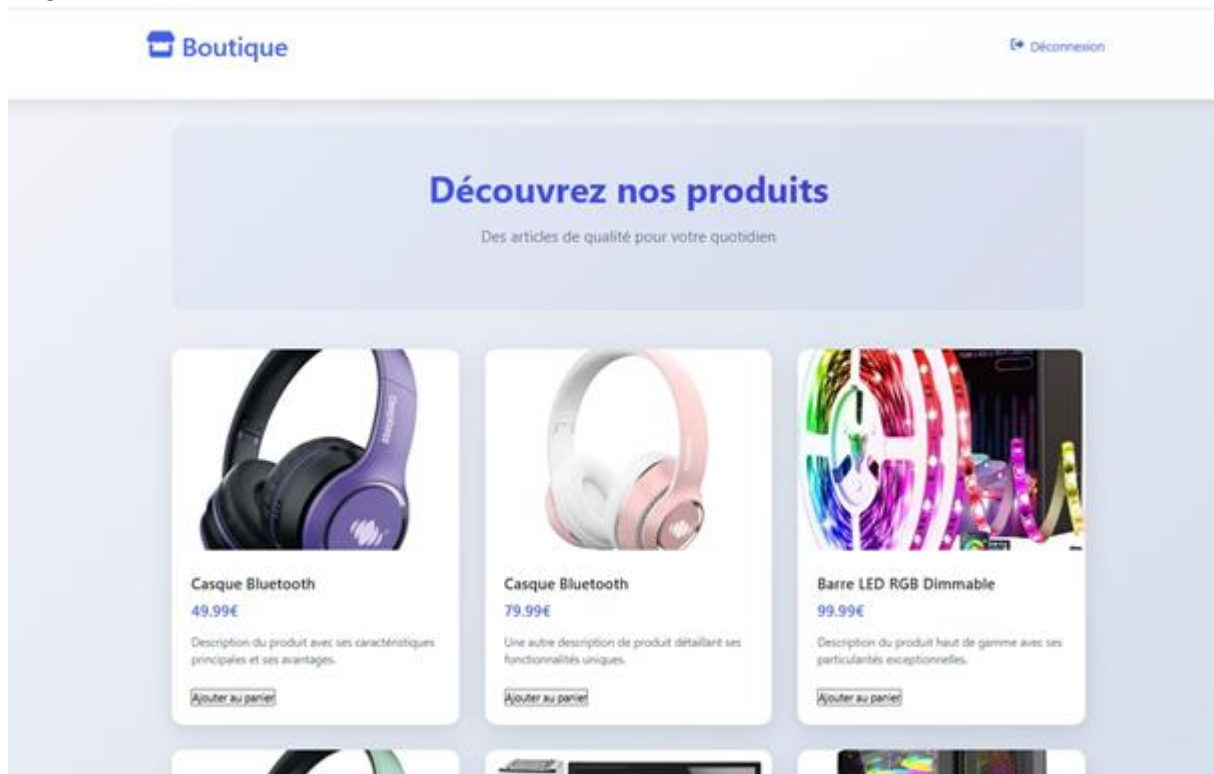


Cette procédure garantit que le site web est correctement déployé et accessible via l'infrastructure réseau tout en respectant les bonnes pratiques de sécurité du serveur web.

Voici notre site web utilisé pour les tests :  
Page login:



Page Home:



### Activation de ModSecurity (WAF) :

ModSecurity est un **Web Application Firewall (WAF)** open source qui fournit une protection en temps réel contre les attaques web courantes. Il agit comme un bouclier entre les applications web et les utilisateurs en analysant les requêtes HTTP/HTTPS et en bloquant les tentatives d'exploitation de vulnérabilités.

L'intégration de ModSecurity dans Apache2 renforce significativement la sécurité de la DMZ en protégeant contre les injections SQL, les attaques XSS (Cross-Site Scripting), les tentatives de traversée de répertoires, et autres menaces web sophistiquées.

\$ sudo apt install libapache2-mod-security2 -y

```
webserver@webserver:~$ sudo apt install libapache2-mod-security2 -y
[sudo] password for webserver:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  liblua5.1-0 libyajl2 modsecurity-crs
Suggested packages:
  lua geoip-database-contrib ruby python
```

## Activation du WAF:

```
GNU nano 7.2 /etc/modsecurity/modsecurity.conf-recommended *
# -- Rule engine initialization -----
# Enable ModSecurity, attaching it to every transaction. Use detection
# only to start with, because that minimises the chances of post-installation
# disruption.
#
SecRuleEngine On_

webserver@webserver:~$ sudo cp /etc/modsecurity/modsecurity.conf-recommended /etc/modsecurity/modsecurity.conf
```

## Restart Apache2 server :

\$ sudo systemctl restart apache2

```
webserver@webserver:~$ sudo systemctl restart apache2
webserver@webserver:~$
```

## Installation des règles OWASP CRS :

L'OWASP Core Rule Set (CRS) est un ensemble de règles de sécurité généralistes qui détectent et bloquent une large gamme d'attaques contre les applications web. Ces règles constituent la base de protection standard contre les vulnérabilités du Top 10 OWASP, incluant les injections, les attaques par force brute, et les tentatives d'exploitation de failles de sécurité.

\$ cd /etc/modsecurity/

\$ sudo git clone <https://github.com/coreruleset/coreruleset.git>

```
webserver@webserver:~$ cd /usr/share
webserver@webserver:/usr/share$ sudo git clone https://github.com/coreruleset/coreruleset.git
Cloning into 'coreruleset'...
remote: Enumerating objects: 35615, done.
remote: Counting objects: 100% (584/584), done.
remote: Compressing objects: 100% (263/263), done.
remote: Total 35615 (delta 539), reused 321 (delta 321), pack-reused 35031 (from 4)
Receiving objects: 100% (35615/35615), 10.58 MiB | 15.26 MiB/s, done.
Resolving deltas: 100% (28173/28173), done.
webserver@webserver:/usr/share$
webserver@webserver:/usr/share$ cd coreruleset
webserver@webserver:/usr/share/coreruleset$ sudo cp crs-setup.conf.example crs-setup.conf
webserver@webserver:/usr/share/coreruleset$
```

Incluons maintenant ces règles dans la configuration de ModSecurity pour Apache:

```
webserver@webserver:/usr/share/coreruleset$ sudo nano /etc/apache2/mods-enabled/security2.conf
```

Ajoutons ces lignes dans le fichier de configuration:

```
GNU nano 7.2 /etc/apache2/mods-available/security2.conf
<IfModule security2_module>
  SecDataDir /var/cache/modsecurity
  Include /usr/share/modsecurity-crs/crs-setup.conf
  Include /usr/share/modsecurity-crs/rules/*.conf
  # Default Debian dir for modsecurity's persistent data
  SecDataDir /var/cache/modsecurity

  # Include all the *.conf files in /etc/modsecurity.
  # Keeping your local configuration in that directory
```

Restart apache2

\$ sudo systemctl restart apache2

```
webserver@webserver:~$ sudo systemctl restart apache2
webserver@webserver:~$
```

Cette configuration active un système de protection robuste contre les attaques web les plus courantes, transformant le serveur Apache en une véritable forteresse de sécurité pour la DMZ.

#### **d. Configuration du Serveur Stockage :**

Dans un contexte d'innovation pour notre projet, une solution de stockage centralisé a été mise en place pour permettre la sauvegarde automatisée des données critiques du réseau, notamment les configurations des équipements, les logs système, et les rapports d'analyse de sécurité.

Cette section décrit les étapes clés de la configuration du stockage, l'automatisation des sauvegardes, ainsi que les outils utilisés.

##### **Création du volume logique :**

La gestion du stockage a été effectuée à l'aide de LVM (Logical Volume Manager), ce qui permet une grande flexibilité dans la gestion des espaces disques. Voici un extrait de la commande utilisée pour afficher les informations sur le groupe de volumes :*sudo vgdisplay*

- VG Name : ubuntu-vg
- VG Size : <148.00 GiB
- PE Size : 4.00 MiB
- Free PE / Size : 18944 / 74.00 GiB



```

adminstock@stockage:~$ sudo vgdisplay
--- Volume group ---
VG Name                ubuntu-vg
System ID
Format                  lvm2
Metadata Areas          1
Metadata Sequence No    2
VG Access                read/write
VG Status                resizable
MAX LV                  0
Cur LV                  1
Open LV                  1
Max PV                   0
Cur PV                  1
Act PV                   1
VG Size                  <148.00 GiB
PE Size                  4.00 MiB
Total PE                 37887
Alloc PE / Size          18943 / <74.00 GiB
Free PE / Size           18944 / 74.00 GiB
VG UUID                  Vh9e8q-r7Pq-OSLW-L7NA-jfXc-kCzs-L0soKE

adminstock@stockage:~$ sudo lvcreate -L 70G -n lv_storage ubuntu-vg
Logical volume "lv_storage" created.
adminstock@stockage:~$ sudo mkfs.ext4 /dev/ubuntu-vg/lv_storage
mke2fs 1.47.0 (5-Feb-2023)
Creating filesystem with 18350080 4k blocks and 4587520 inodes
Filesystem UUID: 68aad01f-7d93-4502-a67f-13127ebfc06f
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
    4096000, 7962624, 11239424

Allocating group tables: done
Writing inode tables: done
Creating journal (131072 blocks): done
Writing superblocks and filesystem accounting information: done

adminstock@stockage:~$

```

Un nouveau volume logique a été créé :

```
sudo lvcreate -n lv_storage ubuntu-vg -L 74G
```

Puis formaté avec le système de fichiers ext4 :

```
sudo mkfs.ext4 /dev/ubuntu-vg/lv_storage
```

### **Montage et persistance du stockage :**

Pour que le volume soit accessible après redémarrage, une entrée a été ajoutée dans le fichier '/etc/fstab'. Exemple d'entrée :

```
UUID=68aad01f-7d93-4502-a67f-13127ebfc06f /mnt/storage ext4 defaults 0 2
```

```

adminstock@stockage:~$ ssh admin@192.168.29.5
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.11.0-26-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Mon Jun  9 16:20:26 2025 from 192.168.29.6
admin@dapservers-VMware-Virtual-Platform:~$

```

le contenu du fichier /etc/fstab dans l'éditeur nano, montrant l'entrée de montage

persistant : UUID=68aad01f-7d93-4502-a67f-13127ebfc06f /mnt/storage ext4  
defaults 0 2

```
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/ubuntu-vg/ubuntu-lv during curtin installation
/dev/disk/by-id/dm-uuid-LVM-Vh9e8qr7Pq0SLWL7NAjfXckCzsL0soKEGLLxdR0u9Fbquaxku3aHPPVP17jD1F2n / ext4 defaults 0 1
# /boot was on /dev/sda2 during curtin installation
/dev/disk/by-uuid/25ad1c3b-9ffa-4762-b88b-d1c9ebee0e1b /boot ext4 defaults 0 1
/swap.img none swap sw 0 0
UUID=68aad01f-7d93-4502-a67f-13127ebfc06f /mnt/storage ext4 defaults 0 2
```

### Automatisation des sauvegardes :

Afin de simplifier la gestion des sauvegardes, un script Bash a été développé et déployé sur le serveur de stockage. Ce script de sauvegarde /usr/local/bin/backup\_to\_server.sh dans nano, contenant la commande rsync avec les exclusions pour synchroniser vers /mnt/storage/machines/client1/full\_backup/

### Script de nettoyage automatique :

l'édition du script /usr/local/bin/compress\_storage.sh avec la commande find pour compresser les fichiers de plus de 1MB dans /mnt/storage/machines.

```
GNU nano 7.2 /usr/local/bin/compress_storage.sh
#!/bin/bash
find /mnt/storage/machines -type f ! -name "*.gz" -size +1M -exec gzip {} ;
echo "Compression faite le $(date)" >> /var/log/storage_maintenance.log_
```

Le script /usr/local/bin/clean\_storage.sh contenant la commande find pour nettoyer les fichiers de plus de 30 jours avec logging vers /var/log/storage\_maintenance.log :

```
adminstock@stockage:~$ sudo chmod +x /usr/local/bin/clean_storage.sh
adminstock@stockage:~$ sudo crontab -e
no crontab for root - using an empty one

Select an editor. To change later, run 'select-editor'.
 1. /bin/nano          <---- easiest
 2. /usr/bin/vim.basic
 3. /usr/bin/vim.tiny
 4. /bin/ed

Choose 1-4 [1]: _
```

### Synchronisation avec le serveur distant :

Une sauvegarde automatique vers un serveur distant a été configurée grâce à 'rsync' et aux clés SSH.

## ➤ Étapes réalisées :

### 1. Génération d'une paire de clés SSH :

L'exécution de `sudo ssh-keygen -t rsa -b 4096 -C "backup-server-key"` avec la génération complète de la paire de clés SSH, incluant :

La création de la clé privée dans `/root/.ssh/id_rsa`

La création de la clé publique dans `/root/.ssh/id_rsa.pub`

L'affichage de l'empreinte de la clé et de l'art ASCII de randomisation

```
adminstock@stockage:~$ sudo ssh-keygen -t rsa -b 4096 -C "backup-server-key"
[sudo] password for adminstock:
Sorry, try again.
[sudo] password for adminstock:
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:r44e0b9gqIEwTp+ebT5p81GZ/PK0hE0X86EgZ2e1pkw backup-server-key
The key's randomart image is:
+----[RSA 4096]-----+
|
|      .+ E +
|     . * * B .
|    . = S . = .
|   +O . . . = .
|  +...O..Bo = .
|   .+O*..* = .
|  .O*=+..+ = .
+-----[SHA256]-----+
adminstock@stockage:~$
```

### 2. Copie de la clé publique sur le serveur cible :

L'exécution de `ssh-copy-id` pour copier la clé publique vers `admin@192.168.29.5`, avec :

La tentative de connexion et d'installation de la clé

Les avertissements de sécurité concernant l'ajout de nouvelles clés au système distant

La confirmation que la clé a été ajoutée avec succès

```
adminstock@stockage:~$ ssh-copy-id -i ~/.ssh/id_rsa.pub admin@192.168.29.5
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: '/home/adminstock/.ssh/id_rsa.pub'
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: WARNING: All keys were skipped because they already exist on the remote system.
(if you think this is a mistake, you may want to use -f option)
adminstock@stockage:~$
```

### 3. Script de sauvegarde :

```
#!/bin/bash
```

```
rsync -a --delete \
```

```
--exclude={"/dev/*","/proc/*","/sys/*","/tmp/*","/run/*","/mnt/*","/media/*"} \
```

```
/ /mnt/storage/machines/client1/full_backup/
```

```
GNU nano 7.2 /usr/local/bin/backup_to_server.sh
#!/bin/bash

rsync -aXv --delete \
--exclude={"/dev/*","/proc/*","/sys/*","/tmp/*","/run/*","/mnt/*","/media/*","/lost+found"} \
/ ldapserver@192.168.29.6:/mnt/storage/machines/client1/full_backup/
```

#### 4. Programmation via 'cron' :

La sélection de l'éditeur pour configurer crontab avec `sudo crontab -e`, étape nécessaire avant d'ajouter la tâche programmée pour l'exécution quotidienne à 2h00 :

```
crontab -e
```

```
# Exécution quotidienne à 2h00
```

```
0 2 * * * /usr/local/bin/backup_to_server.sh >> /var/log/storage_maintenance.log 2>
```

```
adminstock@stockage:~$ sudo chmod +x /usr/local/bin/clean_storage.sh
adminstock@stockage:~$ sudo crontab -e
no crontab for root - using an empty one
```

```
Select an editor. To change later, run 'select-editor'.
```

1. /bin/nano <---- easiest
2. /usr/bin/vim.basic
3. /usr/bin/vim.tiny
4. /bin/ed

```
Choose 1-4 [1]: _
```

## Test de sécurité

---

### 1. Tests du WAF (ModSecurity)

Dans le cadre de ce projet, nous avons mis en place un pare-feu applicatif (WAF) à l'aide de ModSecurity, couplé aux règles OWASP CRS, afin de renforcer la sécurité du serveur web Apache2. Pour valider son efficacité, nous avons effectué des tests sur des attaques courantes : XSS (Cross-Site Scripting) et SQL injection. Ces tests visaient à évaluer si ModSecurity était capable de détecter et d'intercepter ces tentatives d'intrusion.

#### Attaque XSS :

L'attaque XSS consiste à injecter un code JavaScript malveillant dans une page web via les paramètres d'une URL ou les champs de formulaire. L'objectif est souvent de voler des informations sensibles (comme les cookies de session) ou de rediriger l'utilisateur vers un site tiers.

#### ➤ Test réalisé :

URL testée : `http://192.168.142.138/login.html?test=<script>alert(1)</script>`

Description : Injection d'un script JavaScript simple (`<script>alert(1)</script>`) dans un paramètre de l'URL.

Résultat attendu : Le navigateur exécute le script et affiche une boîte de dialogue avec le message `alert(1)`.

### ➤ Résultat obtenu :

Réponse du serveur :



### ➤ Analyse :

- Le script JavaScript n'a pas été exécuté.
- La requête a été bloquée par ModSecurity, probablement grâce à une règle OWASP CRS dédiée aux attaques XSS.
- Cela prouve que le WAF fonctionne correctement pour empêcher l'exécution de scripts non autorisés.

### Test SQL Injection :

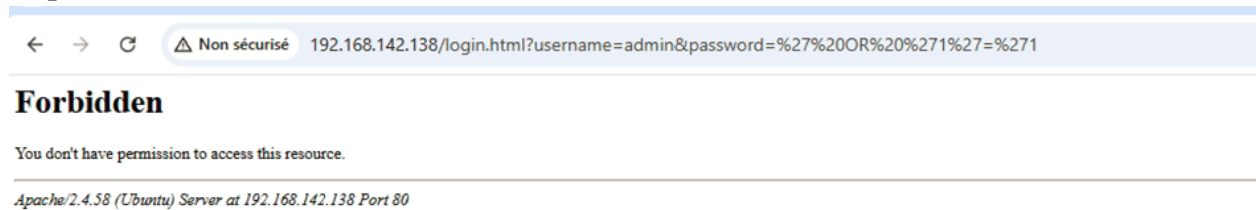
L'attaque SQL injection vise à injecter des commandes SQL malveillantes dans les requêtes HTTP (généralement via les formulaires de connexion). Cela permet d'exploiter des vulnérabilités dans l'application pour accéder à la base de données ou modifier ses contenus.

### ➤ Test réalisé :

- URL testée : <http://192.168.142.138/login.html?username=admin&password='OR '1'='1>
- Description : Injection d'une chaîne SQL qui pourrait falsifier la condition d'authentification ('OR '1'='1').
- Résultat attendu : Si la base de données est vulnérable, l'utilisateur pourrait se connecter sans fournir le mot de passe correct.

### ➤ Résultat obtenu :

Réponse du serveur :



### ➤ Analyse :

- La tentative d'injection SQL a été interceptée par ModSecurity.
- Aucune connexion frauduleuse n'a eu lieu.
- Les règles OWASP CRS ont correctement identifié la séquence 'OR '1'='1' comme une tentative d'attaque SQL injection.
- Cela confirme que le pare-feu applicatif est bien configuré et opérationnel pour bloquer ce type de menaces.

### Conclusion sur les tests du WAF :

Les tests réalisés montrent clairement que ModSecurity, combiné aux règles **OWASP Core Rule Set**, est efficace pour protéger le serveur web contre deux types d'attaques courantes : XSS et SQL injection. Grâce à ces règles préconfigurées, le WAF bloque automatiquement les requêtes suspectes avant qu'elles ne puissent affecter l'application ou la base de données.

Ces résultats soulignent l'importance de mettre en place un pare-feu applicatif dans un environnement web, notamment lorsqu'il s'agit de gérer des données sensibles ou des interactions utilisateur. De plus, ils démontrent la pertinence de notre approche de sécurité basée sur la défense en profondeur, où chaque couche du réseau (IDS/IPS, pare-feu applicatif, etc.) joue un rôle complémentaire dans la protection globale du système.

Pour améliorer davantage la robustesse du WAF, il serait possible de :

- Personnaliser les règles ModSecurity pour mieux correspondre au comportement normal de l'application.
- Activer les journaux d'audit pour suivre précisément les attaques bloquées.
- Utiliser des outils comme **Kibana** ou **ELK Stack** pour visualiser les alertes générées par ModSecurity.



## 2. Test de la méthode de détection d'intrusion avec notre SIEM OSSIM.

Nous lançons un scan du site web depuis notre machine d'attaque Kali Linux, en utilisant l'outil Nikto pour détecter les vulnérabilités

```
~/Desktop$ nikto -h 192.168.29.129
```

Résultats du scan :

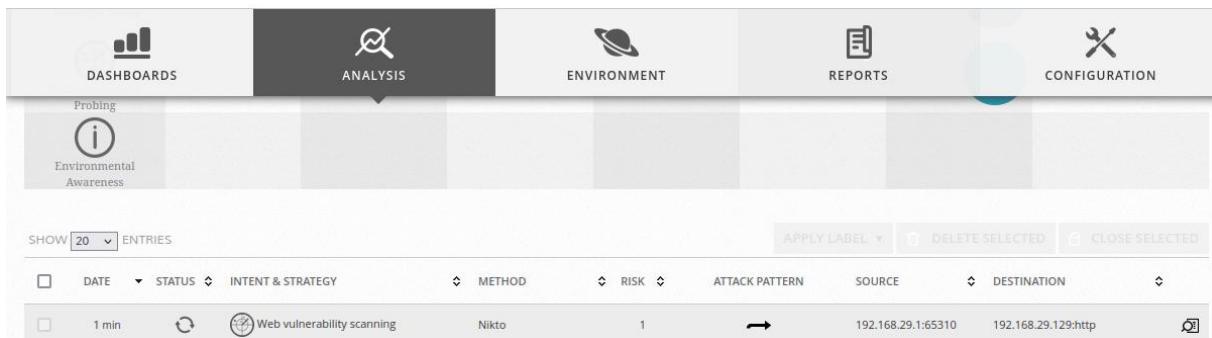
- **Target IP :** 192.168.29.129  
→ Adresse IP de la machine cible (locale).
- **Target Hostname :** 192.168.29.129  
→ Nom d'hôte (identique à l'adresse IP ici).
- **Target Port :** 80  
→ Port analysé, ici le port HTTP classique.
- **Start Time / End Time :**  
→ Le scan a commencé à 20:40:47 et s'est terminé à 20:41:15, donc il a duré 28 secondes.
- **Server :** Apache/2.4.58 (Ubuntu)  
→ Le serveur web est Apache version 2.4.58 sur un système Ubuntu.

```
- Nikto v2.1.5
-----
+ Target IP:      192.168.29.129
+ Target Hostname: 192.168.29.129
+ Target Port:    80
+ Start Time:     2025-06-10 20:40:47 (GMT1)
-----
+ Server: Apache/2.4.58 (Ubuntu)
+ The anti-clickjacking X-Frame-Options header is not present.
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ 6544 items checked: 0 error(s) and 1 item(s) reported on remote host
+ End Time:       2025-06-10 20:41:15 (GMT1) (28 seconds)
-----
+ 1 host(s) tested
```

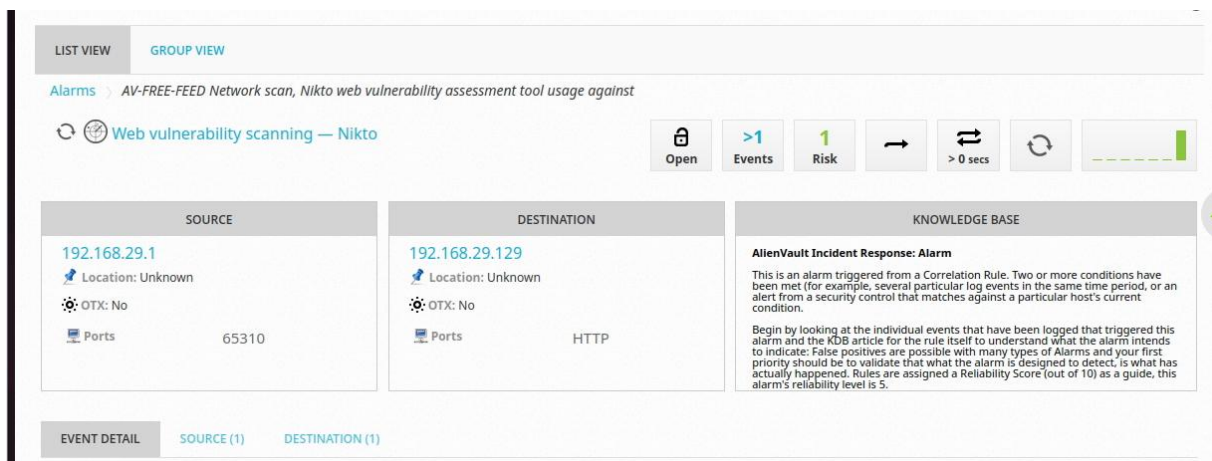


## Notre système IDS/SIEM OSSIM

On peut voir que notre système OSSIM affiche une alerte de sécurité



Il affiche également des informations telles que la méthode de scan utilisée.



### 3. Analyse et discussion

Nous présentons ici une analyse critique de l'ensemble du travail effectué. Cette section vise à évaluer les forces, les faiblesses, les difficultés rencontrées, ainsi que les perspectives d'amélioration possibles de notre architecture réseau et des solutions de sécurité mises en œuvre.

#### 3.7.1 Forces du projet :

##### ➤ **Architecture réaliste et bien segmentée :**

L'infrastructure conçue reflète fidèlement un environnement professionnel, avec une segmentation claire entre :

- **Internet**
- **DMZ** (hébergeant le serveur web Apache protégé par ModSecurity)
- **LAN interne** (réservé aux postes clients)
- **Réseau de gestion et de surveillance** (incluant AlienVault OSSIM)

Cette segmentation permet une isolation efficace des services critiques tout en facilitant la mise en place de politiques de sécurité adaptées.

➤ **Diversité des outils utilisés :**

Nous avons utilisé une panoplie d'outils professionnels et open source :

- **GNS3** : pour la modélisation du réseau
- **FortiGate** : pour la protection périmétrique
- **Apache + ModSecurity + OWASP CRS** : pour la sécurité applicative
- **AlienVault OSSIM** : pour la centralisation des logs et la corrélation des événements
- **LDAP** : pour la gestion centralisée des identifiants utilisateurs
- **Stockage LVM + rsync + cron** : pour la sauvegarde automatisée

La combinaison de ces technologies illustre une défense en profondeur efficace et conforme aux bonnes pratiques en cybersécurité.

➤ **Détection proactive des menaces**

Grâce à ModSecurity, nous avons pu simuler une réponse rapide face à divers types d'attaques :

- Scan de ports via Nikto
- Injection SQL sur le serveur web
- XSS (Cross-Site Scripting)

Les alertes générées ont permis de valider l'efficacité de nos règles de détection, tant au niveau réseau qu'au niveau applicatif.

➤ **Centralisation des logs et visibilité améliorée**

Avec AlienVault OSSIM, nous avons mis en place une plateforme de centralisation des journaux qui agrège les événements provenant de différents équipements (pare-feu, serveur web, systèmes Linux). Cela a permis une corrélation des incidents, une visualisation consolidée, et une meilleure traçabilité des actions malveillantes.

**3.7.2. Faiblesses et limites observées :**

➤ **Gestion des faux positifs :**

L'un des défis majeurs a été la gestion des faux positifs. Certaines requêtes légitimes ont été bloquées ou signalées comme suspectes, ce qui pourrait conduire à des interruptions inutiles ou à une perte de temps lors d'une utilisation réelle.

➤ **Performances et ressources :**

ModSecurity peut ralentir le traitement des requêtes si les règles ne sont pas optimisées.

### ➤ **Complexité de la configuration initiale :**

La mise en place de certains composants tels que le serveur LDAP, la synchronisation SSH/rsync, ou encore l'intégration des règles personnalisées dans ModSecurity a demandé un temps d'apprentissage et de débogage non négligeable.

### ➤ **Limitations de la simulation**

Bien que GNS3 offre une grande flexibilité, certaines fonctionnalités matérielles ou comportementales (comme la latence réelle, les performances réseau, ou les failles physiques) ne peuvent pas être simulées parfaitement dans un environnement virtuel.

### **Difficultés rencontrées :**

- **Configuration des interfaces VLAN et routage inter-VLAN :** il a fallu s'assurer que les communications entre zones soient bien contrôlées.
- **Intégration des pare-feu FortiGate :** la configuration manuelle des règles de filtrage a nécessité une bonne compréhension des flux réseau.
- **Déploiement de ModSecurity :** les règles OWASP CRS sont nombreuses, mais nécessitent souvent d'être affinées pour éviter les blocages inutiles.
- **Centralisation des logs avec OSSIM :** certains agents ou plugins n'étaient pas immédiatement compatibles, ce qui a nécessité une adaptation locale.
- **Synchronisation des sauvegardes :** le script rsync a dû être testé plusieurs fois pour gérer correctement les exclusions de répertoires sensibles.

### **Perspectives d'amélioration :**

### ➤ **Intégration d'un SIEM avancé :**

AlienVault OSSIM est une solution robuste, mais elle pourrait être remplacée ou complétée par des outils plus avancés comme ELK Stack (Elasticsearch, Logstash, Kibana) ou Splunk, offrant une analyse plus poussée et une visualisation interactive des données de sécurité.

### ➤ **Automatisation des réponses aux incidents :**

Actuellement, les alertes sont centralisées, mais la réponse reste manuelle. Une évolution possible serait d'intégrer des scripts ou des playbooks Ansible pour automatiser le blocage des IP suspectes ou la mise à jour des règles du pare-feu.

### ➤ **Utilisation de règles personnalisées et d'apprentissage automatique :**

Des algorithmes d'apprentissage automatique pourraient être explorés pour renforcer la détection comportementale, notamment pour identifier des attaques zero-day ou polymorphes.

➤ **Amélioration de la gestion des utilisateurs :**

Le serveur LDAP fournit une base solide pour la gestion des accès, mais il pourrait être enrichi avec :

- Une **authentification multi-facteurs (MFA)**
- Une **gestion centralisée des accès SSH**
- Une **intégration avec Active Directory**

➤ **Optimisation du stockage et des sauvegardes :**

La mise en œuvre d'une journalisation horodatée, de versions incrémentielles, ou de snapshots de volumes permettrait une meilleure gestion des restaurations rapides en cas de corruption ou d'attaque.

Ce projet a permis de consolider nos compétences techniques en matière de conception, déploiement et sécurisation d'un réseau informatique complet. Grâce à la mise en œuvre de technologies clés telles que ModSecurity, FortiGate, et OSSIM, nous avons pu expérimenter concrètement les principes de la cybersécurité :  
Segmentation du réseau Surveillance proactive Centralisation des logs  
Authentification centralisée Protection applicative Malgré quelques limitations liées à la simulation et à la complexité technique, cette étude constitue une expérience précieuse et proche de la réalité opérationnelle.

## Conclusion

---

Ce projet nous a permis de mettre en pratique nos connaissances en cybersécurité dans un environnement simulé, proche de la réalité professionnelle. En utilisant l'outil GNS3, nous avons conçu une infrastructure réseau complète, incluant des équipements tels que des switches L3, des pare-feu FortiGate, un serveur web sécurisé, ainsi qu'un serveur LDAP pour la gestion centralisée des utilisateurs.

Grâce à cette approche pédagogique, nous avons pu expérimenter les bonnes pratiques en matière de sécurité informatique, notamment :

- La segmentation du réseau pour isoler les zones sensibles (LAN, DMZ) ;
- La mise en place d'un pare-feu applicatif (WAF) via ModSecurity pour protéger le serveur web contre des attaques telles que les injections SQL ou les failles XSS ;
- L'utilisation d'un système de centralisation des journaux avec AlienVault OSSIM, facilitant la corrélation des événements et la réponse aux incidents ;
- La configuration d'un serveur LDAP pour gérer les identifiants utilisateurs de manière centralisée ;
- L'intégration d'un système de stockage et de sauvegarde automatisé, assurant la disponibilité et la pérennité des données critiques.

Au-delà de l'aspect technique, ce projet a également renforcé nos compétences en :

- **Travail d'équipe** : répartition efficace des tâches selon les forces de chacun ;
- **Gestion de projet** : organisation des phases clés (conception, déploiement, tests), respect des délais, coordination entre les membres ;
- **Configuration réseau avancée** : routage inter-VLAN, paramétrage des interfaces, gestion des règles de filtrage ;
- **Analyse de sécurité** : surveillance du trafic, lecture des logs, détection des comportements anormaux.

En somme, ce projet a été une expérience riche, tant sur le plan académique que professionnel. Il nous a permis de mieux comprendre les enjeux de la cybersécurité dans un contexte réseau complexe, tout en développant une vision globale et opérationnelle des outils et méthodes utilisés dans ce domaine.

## Références bibliographiques :

---

OWASP ModSecurity CRS

- Documentation officielle : <https://coreruleset.org/>

GNS3 Documentation

- Documentation officielle : <https://docs.gns3.com/>

Fortinet Documentation Library. (2023). *FortiGate Administration Guide – v7.2*.

- Guide complet sur la configuration des interfaces, politiques, objets réseau et fonctions de sécurité : <https://docs.fortinet.com>

OpenLDAP Project. (2022). *OpenLDAP Administrator's Guide*.

- Documentation officielle sur la gestion des services slapd, fichiers LDIF, et authentification LDAP. : <https://www.openldap.org/doc/admin24/>

AlienVault OSSIM Documentation. (2023). *OSSIM User Guide*.

- Guide de l'utilisateur pour l'installation, configuration des sources de logs, et gestion des alertes : <https://cybersecurity.att.com/products/ossim>

Apache Software Foundation. (2023). *Apache HTTP Server Documentation – Version 2.4*.

- Installation, configuration et déploiement de contenu web avec Apache2 : <https://httpd.apache.org/docs/2.4/>